1989

Signal processing and preliminary results in the 1988 Monterey Bay Tomography Experiment.

Dees, Robert Charles.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Signal Processing and Preliminary Results in the
1988 Monterey Bay Tomography Experiment

by

Robert C. Dees

June 1989

Thesis Advisor: James H. Miller

Approved for public release; distribution unlimited.

Security Classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification: Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report: Approved for public release; distribution is unlimited.
2b Declassification/Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization: Naval Postgraduate School; 6b Office Symbol (If Applicable): 62Mr	7a Name of Monitoring Organization: Naval Postgraduate School
6c Address (city, state, and ZIP code): Monterey, CA 93943-5000	7b Address (city, state, and ZIP code): Monterey, CA 93943-5000
8a Name of Funding/Sponsoring Organization; 8b Office Symbol (If Applicable)	9 Procurement Instrument Identification Number
8c Address (city, state, and ZIP code)	10 Source of Funding Numbers: Program Element Number; Project No; Task No; Work Unit Accession No

11 Title (Include Security Classification): SIGNAL PROCESSING AND PRELIMINARY RESULTS IN THE 1988 MONTEREY BAY TOMOGRAPHY EXPERIMENT

12 Personal Author(s): Robert C. Dees

13a Type of Report	13b Time Covered	14 Date of Report (year, month, day)	15 Page Count
Master's Thesis	From To	June 1989	190

16 Supplementary Notation: The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (continue on reverse if necessary and identify by block number)
Field	Group	Subgroup	Acoustic tomography, Fast Hadamard Transform, maximal-length sequences, internal waves, surface wave spectra.

19 Abstract (continue on reverse if necessary and identify by block number)

Ocean acoustic tomography is particularly suited to observing mesoscale dynamic processes, which may not be adequately observed by more conventional methods. Ships and buoys are limited in their sampling rates by location and/or transit speed while the tomographic signal samples the current and temperature fields all along its path at the speed of sound. Variation in the travel time of the signal occurs due to inhomogeneity in either the sound speed or the current. The ocean's fluctuation can then be estimated from the travel time perturbation using mathematical inverse methods. The 1988 Monterey Bay Tomography Experiment had several specific goals: to test new technology for real-time transmission of tomographic data to shore, to examine the feasibility of doing acoustic tomography in a coastal environment, and to examine the effects of coastal ocean processes such as surface and internal waves and a rough bottom on the tomography signal.This thesis concentrates on signal design using maximal-length sequences, data recording, and a fast algorithm for a data-synchronous digital correlator receiver in this experiment.The new tomographic data recording system has demonstrated its effectiveness. Preliminary results of the data analysis are given , including power spectra for the arrival time perturbation series in the 0.01 to 0.26 Hz (surface wave) frequency band. These spectra correlate well with surface wave spectra obtained from a wave-measuring buoy. Low-pass filtered time series showing perturbations at internal wave frequencies are also presented.

20 Distribution/Availability of Abstract	21 Abstract Security Classification	
[X] unclassified/unlimited [ ] same as report [ ] DTIC users	Unclassified	
22a Name of Responsible Individual: James H. Miller	22b Telephone (Include Area code): (408) 646-2384	22c Office Symbol: 62Mr

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted — All other editions are obsolete

Approved for public release; distribution is unlimited.

Signal Processing and Preliminary Results in the 1988 Monterey Bay Tomography Experiment

by

Robert Charles Dees
Lieutenant, United States Navy
B.S., University of Kansas, 1982

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING ACOUSTICS

from the

NAVAL POSTGRADUATE SCHOOL
June 1989

ABSTRACT

Ocean acoustic tomography is particularly suited to observing mesoscale dynamic processes, which may not be adequately observed by more conventional methods. Ships and buoys are limited in their sampling rates by location and/or transit speed while the tomographic signal samples the current and temperature fields all along its path at the speed of sound. Variation in the travel time of the signal occurs due to inhomogeneity in either the sound speed or the current. The ocean's fluctuation can then be estimated from the travel time perturbation using mathematical inverse methods. The 1988 Monterey Bay Tomography Experiment had several specific goals: to test new technology for real-time transmission of tomographic data to shore, to examine the feasibility of doing acoustic tomography in a coastal environment, and to examine the effects of coastal ocean processes such as surface and internal waves and a rough bottom on the tomography signal. This thesis concentrates on signal design using maximal-length sequences, data recording, and a fast algorithm for a data-synchronous digital correlator receiver in this experiment. The new tomographic data recording system has demonstrated its effectiveness. Preliminary results of the data analysis are given , including power spectra for the arrival time perturbation series in the 0.01 to 0.26 Hz (surface wave) frequency band. These spectra correlate well with surface wave spectra obtained from a wave-measuring buoy. Low-pass filtered time series showing perturbations at internal wave frequencies are also presented.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

It is particularly difficult to put together a list of those people who have contributed to this thesis. Because so many people have helped me with the work, any boundary between those listed and those not is artificial and hard to set. If I could, I would express my gratitude personally to everyone who has expressed confidence in me. Nevertheless, I will try to acknowledge those most involved and sincerely hope that no one will feel slighted. To start, I thank the three principle investigators in the experiment: Dr. James H. Miller, Dr. Ching-Sang Chiu, and Dr. James F. Lynch. They are three very different men whose abilities and interests complement each others, allowing them to form an excellent research team. Their patience and encouragement has taught me a great deal about ocean acoustics. I thank Floyd York, Sönke Paulsen, Kevin Schaaf, Paul Boutin, and the crew of the R/V Point Sur for making the experiment work. I thank one of my professors, Dr. Anthony A. Atchley, who was exceptional in the amount of understanding he could impart through his teaching in acoustics. A special thanks to Dr. Kurt Metzger for his early encouraging opinions on the viability of the new data acquisition system. And, for their very early work, I thank my parents, John and Barbara Dees.

I. INTRODUCTION

A. A SUMMARY OF THE THESIS

This thesis describes the 1988 Monterey Bay Acoustic Tomography Experiment and preliminary data analysis, concentrating on the signal processing involved. The goal of the experiment was to test a relatively simple tomographic system for a coastal environment, and to quantify the effects of surface and internal waves on tomographic signals. This experiment was different from other tomography experiments in that the tomographic signal was of short duration (1.9375 seconds) and was transmitted continuously, allowing the effect of surface waves of periods greater than 4 seconds to be measured. The question that was posed was whether the spectra of the surface and internal waves could be mapped from the travel time perturbation series. If this could be done, future tomography systems could have enhanced capabilities. First, they could be used to investigate the surface and internal waves, and second, the noise in measurements of eddy fields could be much better parameterized.

The experiment was conducted from 12-16 December 1988 in Monterey Bay between a single acoustic source located on an unnamed seamount off Point Sur transmitting to ten receivers located around the Monterey Canyon between Point Piños and Santa Cruz. The transmission paths begin with a source depth of 870 meters before crossing various parts of the Monterey Canyon (about 2300 meters depth) and ending on the shallow shelf around the canyon. All the receivers were located in about 100 meters of water.

The thesis is structured as follows:

Chapter II. A discussion of the tomography experiment.

Chapter III. Signal processing used in data collecting.

Chapter IV. Preliminary data analysis.

Chapter V. Conclusions and recommendations for further work.

The 1988 Monterey Bay experiment is the subject of the second chapter. This experiment was primarily targeted at investigating the effects of internal waves and surface waves on the tomography signals. Additionally, a new tomography system for coastal environments was tested. The location of the test in Monterey Bay spans the Monterey submarine canyon where the extreme bathymetry was expected to cause complex ray paths not confined to a two dimensional plane. The effects of the bathymetry and three dimensional ray paths are being investigated but will not be discussed in depth in this thesis. A summary of the oceanographic features of Monterey Bay is also contained in Chapter II as well as a description of measurements made to corroborate the results obtained from tomography. Appendix A contains a chronological summary of the experiment.

The third chapter discusses the signal processing used to convert the raw acoustic data into arrival time perturbation data. Pulse compression is used to achieve signal gain over the maximum that can be attained in single pulses while maintaining the narrow width of the pulse (which is dependent on the transmitter bandwidth). The pulse compression is obtained by a matched-filter correlation for a maximal-length sequence that is phase-modulating the 224 Hz acoustic carrier signal. The hardware involved in analog filtering, the digitization, and for correlation are presented. Methods for code design and

correlation using the Fast Hadamard Transform are included as Appendix B. Program descriptions and listings are contained in Appendix C. The programs utilizing the fast algorithms presented were able to complete the matched-filtering of the received signal about 300 times faster than algorithms using the Discrete Fourier Transform. In order to store the large amount of data from the continuous transmission, the data was transmitted by radio to a shore station and recorded on tapes. During the experiment the signal processing equipment and software were not ready and the pulse-code modulated analog acoustic data was recorded on videocassettes. The system as operating now digitizes and performs the matched-filtering for the maximal-length sequence on two channels in real-time.

Chapter IV presents data from the receiver at Station J as an example. The received signal was digitized and matched-filtered for pulse compression before storage. A convenient method of displaying this data is in waterfall plots so that any signal repeating at the code repetition frequency is easily observed. The arrival time of the pulse is estimated as a difference from an arbitrary start point repeating at the code repetition frequency. Since the code repeats continuously, the absolute travel time of the signal cannot be measured. The fluctuations in the arrival time reflect the dynamic changes in the ocean sound speed field. A comparison between the power spectrum of the arrival time perturbations and the surface wave displacement power spectrum is given. Additional plots are presented in Appendix D.

The final chapter summarizes the work completed in the thesis. The data analysis in the Monterey Bay experiment is only partially completed at the time of writing. The analysis of the sound speed and current measurements is

in progress. The three dimensional ray tracing program is expected to be able to predict eigenrays soon, but still requires modification before it will accurately predict paths with the extremes in bathymetry found in Monterey Bay. The data has already shown the influence of surface waves, something that had not been shown conclusively before. The continuous nature of the data is revealing an interesting internal wave effect, although the analysis is not complete. In short, the data processing and analysis is not finished but has already produced important results and is expected to produce more as work continues.

B. ACOUSTIC TOMOGRAPHY BACKGROUND

"Ocean acoustic tomography is a technique for observing the dynamic behavior of ocean processes by measuring the changes in travel time of acoustic signals transmitted over a number of ocean paths."[Ref. 1] The word tomography is derived from two Greek roots meaning "to slice" and "to look at." Ocean acoustic tomography uses sound energy to look at a "slice" of the ocean by measuring the travel time of signals propagating through the water. Sound speed in the ocean is a function of salinity, pressure, and temperature. As acoustic energy travels along its path, its rate of travel varies with these quantities as well as with the speed and direction of any currents. Mathematical inverse methods are applied to these travel time fluctuations to estimate the variation of these parameters dynamic ocean variables.

Space-based remote sensing has been of enormous importance to physical oceanographers in discovering large scale fluctuations in the ocean. Oceanographers for many years have had difficulty in measuring the general flow patterns because of extremely high "noise" from mesoscale eddy

circulation. These mesoscale features usually last for a few months and have sizes of several hundred kilometers. They are often likened to atmospheric weather. Measurement of these features by satellite remote sensing depends on radiated or reflected electromagnetic radiation and is limited to the surface of the ocean. Interior features may have little or no surface manifestation or may have a complex signature on the surface.[Refs. 2,3,4]

Traditional shipborne measuring systems are limited in their ability to take enough data by their slow speed. To sample mesoscale features one ship would be insufficient; the ship would not finish sampling the area before conditions at the beginning point had changed. The mesoscale features are very important in understanding the oceans. It has been estimated that mesoscale features contain 90 percent of the ocean's kinetic energy.[Ref. 1]

The oceans are relatively transparent to sound. Sound has been used for signaling and tracking in the ocean for many years. Much of what is known about sound in the ocean was discovered as a result of SONAR propagation experimentation, where the goal was to calculate the sound propagation from the known ocean characteristics. Previous experiments would consider how the ocean would affect propagation of the desired signal through the ocean, but it was a tomography experiment conducted off Bermuda in 1981 which first attempted to exploit the properties of the sound field itself as a way of measuring the mesoscale ocean characteristics.[Ref. 2]

Ocean acoustic tomography was originally proposed by Munk and Wunsch in 1977. In 1979, they presented methods for inverting the data to estimate the sound speed field.[Ref. 3] This procedure is similar to the procedure used in medical x-ray tomography where the measuring signal

travels in a straight line from transmitter to receiver. Ocean acoustic tomography may have energy traveling along several curving paths with different travel times and from one transmitter to several receivers simultaneously, as shown in Figure 1. Thus, with several sound sources and receivers, the amount of data collected grows multiplicatively rather than additively (as in point sampling). The sound speed fluctuations along the entire path affect the travel time of a signal. Because of this integrating characteristic of the travel time, small inhomogeneities will have a negligible effect. Sound also has the advantage of sampling the along its path very quickly - approximately 1500 meters per second. If transmissions are made in both directions along a path, the difference in travel time is related to currents along the path.[Ref. 1] Ocean acoustic tomography is a valuable tool for monitoring the ocean interior. Its overall system performance can be improved, however, if it is supplemented by in situ measurements by ships and buoys.

C. THE IMPORTANCE OF THIS EXPERIMENT

This ocean acoustic tomography experiment was unique in two ways:

1. The rapid and continuous repetition of the tomography signal.
2. That the data was transmitted to shore in real-time.

The experiment was conducted in a coastal environment where the shallow water causes ray paths which have many surface interactions. These interactions coupled with the fast repetition of the signal give travel time perturbations whose power spectrum reflects the surface wave power

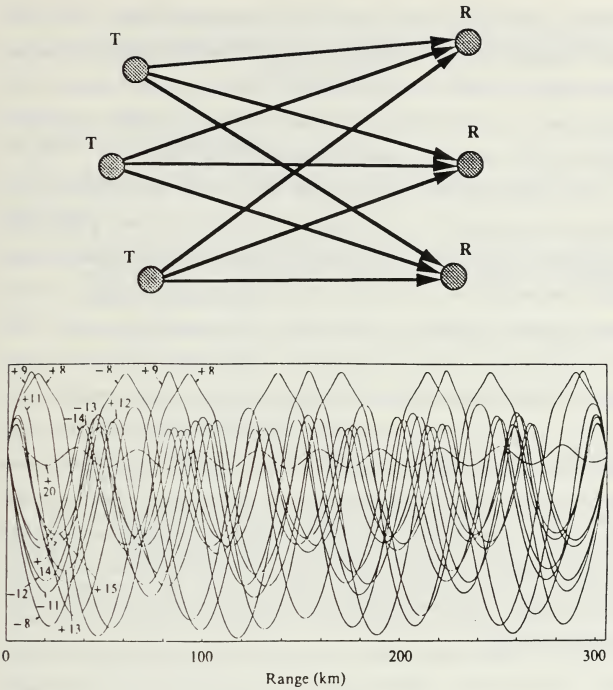


Figure 1: (top) Several transmitters (T) and receivers (R) give many ray paths as viewed from above. (bottom) Each slice may contain numerous eigenrays connecting the transmitters and receivers. This diagram is from a 1983 experiment near Bermuda [Ref. 2].

spectrum, something never conclusively demonstrated before. When the high frequency surface wave signal is filtered out, there remains a strong perturbation at internal wave frequencies. The time perturbations in this experiment were much larger than those seen in tomography experiments with ray paths confined to sound channels in the deep ocean.

The experiment tested a new system for real-time data aquisition. The receivers transmitted the acoustic signal to a shore-based recording system for storage (presently limited to 12 channels in this system). The digitization and pulse-compression for two signals can be conducted in real-time with the simple system described in this thesis, and measuring the travel time perturbation on an arrival could also be added to the program. The data acquisition and processing for a real-time tomography system is within reach. If such a system is built and operated, valuable information on the present state of the ocean could be measured. In some ways this would resemble the "weather radar" used by meteorologists for observing the world as events happen. The value for ocean modeling is enormous. A prediction of SONAR sound propagation in a long-range system could then be improved by using measured data instead of historical data bases and limited information from satellites and ships.

II. THE 1988 MONTEREY BAY ACOUSTIC TOMOGRAPHY EXPERIMENT

A. EXPERIMENT OBJECTIVES

The December 1988 Monterey Bay acoustic tomography experiment had four goals:

1. Investigate the relation between the frequency-direction spectrum of surface waves and the spectra of travel time changes in tomography signals experimentally.
2. Investigate the effect of internal waves on tomography signals in a coastal environment.
3. Investigate the effect of complex three dimensional bathymetry on long range acoustic propagation.
4. Test the first real-time shore-based tomography data acquisition system.

The most significant difference between this experiment and other ocean tomography experiments is in the transmitted signal. For this experiment the signal repeated every 1.9375 seconds, allowing sampling above the Nyquist frequency of dynamic ocean processes with frequencies below 0.258 Hz, which includes the longer period surface gravity waves. Surface gravity waves are classified by their period length: fully developed seas - 5 to 12 seconds, swell - 6 to 22 seconds, and surf beat - 1 to 3 minutes[Ref. 5]. All of these could have observable effects, depending on their orientation to the signal path. The signal was also transmitted continuously. In most other experiments the signal is transmitted periodically to reduce the power consumption and amount of data to be recorded. The continuous transmission permits long

period disturbances to be investigated without the aliasing effects of higher frequency internal waves and surface waves. Aliasing of high frequency energy to low frequencies could be a problem if only a few, time-separated transmissions are used. Internal waves and tidal fluctuations will be of much longer period than the longest swell - periods of 8 minutes (internal waves in shallow water) to 24 hours are possible. The principle investigators for the experiment were Professors James H. Miller and Ching-Sang Chiu of the Naval Postgraduate School and Dr. James F. Lynch of Woods Hole Oceanographic Institution. Both institutions are now jointly working on the data analysis.

B. MONTEREY BAY ENVIRONMENT

The information on the environment of Monterey Bay is presented in much greater detail in a thesis by Theresa Rowan who conducted an initial assessment of receiver placement and ray tracing [Ref. 6].

1. Location and Description

The tomography experiment extended from a transmitter placed on an unnamed seamount 36 kilometers west of Point Sur to receivers placed along the north side of Monterey on the continental shelf between Moss Landing and Santa Cruz. This area and the placement of the transmitter and receivers is detailed in Figure 2. Monterey Bay is a semi-enclosed bay containing a submarine canyon cut into the continental shelf. This canyon (the largest on the California coast) dominates the bathymetry by cutting the bay into two roughly equal halves. The continental shelf surrounding the canyon is fairly smooth with a slope of 1 - 2 percent from shore to a depth of

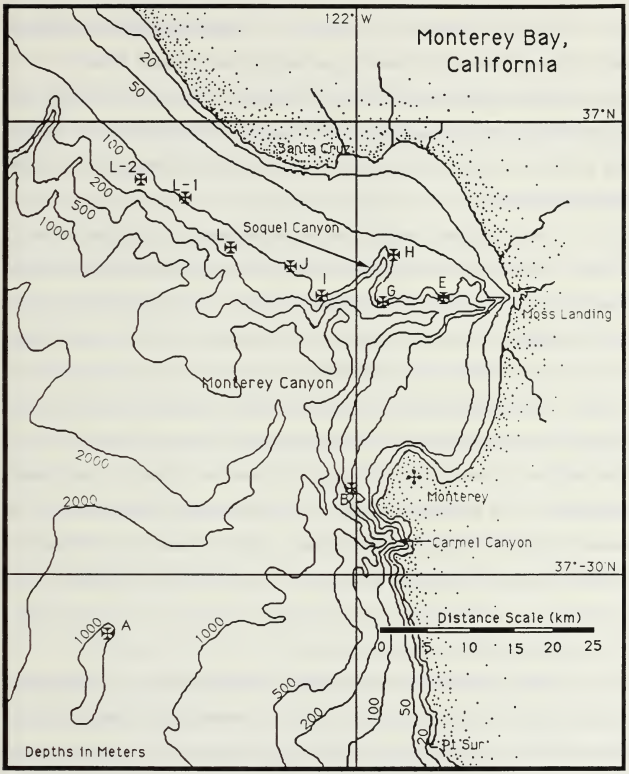


Figure 2: Monterey Bay showing the positions of the transmitter and receivers (positions are marked with ✠). The transmitter is at station A, all others are receivers. The shore station position is marked by ✣ .

90 to 100 meters on the north canyon rim and approximately 180 meters on the south rim. The canyon itself drops sharply from the shelf. The axis of the canyon is steep with a grade of around 7 percent for most of its length and ends in a fan valley at a depth of 2925 meters. Several smaller canyons join the Monterey submarine canyon, most notably the Soquel and Carmel canyons.[Ref. 6]

The initial ray tracing done in preparation for the experiment by Theresa Rowan used a two-dimensional ray-tracing program called Multiple Profile Ray-Tracing Program (MPP)[Ref. 6]. This program used various sound speed profiles and took into consideration the bathymetry along planar paths between source and receiver. The shortcoming of this program is that it neglects horizontal deflection of acoustic energy. Eigenrays which leave a vertical plane between source and receiver can be reflected or refracted back to the receiver in such a complicated environment dominated by rough bathymetry. It is possible that there are stable raypaths which arrive at the receiver by bouncing off the submarine canyon walls via three-dimensional paths which are not close enough to two-dimensional solutions to be identified.

2. Currents, Tides, and Waves

The California Current flowing to the south along the entire west coast of North America brings most of the surface water to Monterey Bay. The California Current brings cold water which is high in nutrients from the Northern Pacific Ocean. This current is broad, extending 700 to 1000

kilometers from the coast along California, but only extending to a depth of less than 500 meters[Ref. 6].

Below the California Current is the California Counter-Current which consists of warm, salty, low-nutrient water moving north from the vicinity of Baja California. The core of this current is often at about 200 meters depth and extends 50 to 100 kilometers offshore. Because of coriolis force and its northward travel the counter-current hugs the shore more tightly than the California Current. Usually, in fall or early winter, conditions occur to bring the counter-current to the surface, where it is called the Davidson Current. This generally happens in Monterey Bay between November and February and is the expected condition for the December experiment[Ref. 6]. This would give a warm surface layer of water above the colder deep water – a stronger density gradient to support internal wave propagation.

The tidal pattern in Monterey Bay is described as a mixed diurnal tide with two high and two low tides of differing heights each day. The range of tide between the higher high tide and the lower low tide is about two meters. The tides occur at all points in the bay with time differences of less than 15 minutes and so the tides in Monterey Bay are only measured with a single tide gauge, at Monterey Fisherman's Wharf.[Refs. 6,7]

The surface waves in Monterey Bay that are of interest in this experiment are the fully developed seas and swell having periods greater than 4 seconds. Climatic information shows that winter (November to March) waves in the Bay generally come from the northwest with periods of 8 to 10 seconds. These waves are the product of storms which may have occurred as far away as the Gulf of Alaska[Ref. 6]. This long period sea swell

was expected to be at the right frequency and travelling perpendicular to the direction of sound propagation which would give the maximum effect for this experiment.

Internal waves result when energy is trapped in oscillations at a density gradient in ocean water. An simple description may help in visualizing internal waves. Consider a "parcel" of water of a certain mass and average density, and that this density is the same as exists at the center of a water column with a constant density gradient. At rest in the center of the column, the parcel is neutrally buoyant and remains at rest. But if it is displaced upward it will be heavier than the surrounding water and will try to sink. As it sinks, the massive water parcel gains velocity (converting its potential energy to kinetic energy) and continues past its neutral buoyancy position before the upward force begins to slow it. This continues as oscillatory motion with the parcel mass moving on the buoyancy spring. Without a density gradient, there will be no restoring force and hence no oscillatory motion. Also, the steeper the gradient, the stiffer the "spring" and the higher the maximum frequency. This is a simplification of the process which actually occurs but serves as a starting point for understanding. The boundary conditions of the density gradient determine which modes of fluctuation can exist[Ref. 4]. Two characteristics of internal wave propagation govern most of the effect seen by tomography:

1. Internal waves have their greatest vertical displacement where the density gradient is maximum[Ref. 8].
2. The maximum frequency which will propagate in the density "waveguide" is a function of the maximum density gradient magnitude [Ref. 4].

The greatest vertical displacement occurs at the sharpest density gradient. This usually coincides with the sharpest sound speed gradient.

The total density gradient between the surface and the bottom of the ocean is often no more than 0.1 percent while the mass which is moved is much greater than that associated with surface wave motion. Consequently, internal waves have much lower frequencies and much lower speeds. Internal waves travel about 100 times more slowly than surface waves[Ref. 4]. Typical periods range from ten minutes in shallow water to 12 hours for tide-driven internal waves.

3. Receiver Placement

The tomography signal receivers for the experiment utilize fixed hydrophones located on the ocean bottom so that receiver motion would not cause arrival time fluctuations. The placement of the receiver was dictated by several requirements. First and most important, the paths of the eigenrays must pass through the water that is of interest in order to sample the sound speed. Second, there should be several eigenrays identifiable passing from the source to the receiver to give vertical resolution. Third, there should be enough separation in the arrival times of the rays traveling along different paths for the received signal to be resolved into distinct arrival times.

The area of interest in this experiment is the Monterey Bay submarine canyon and the edge of the continental shelf along the north rim of the canyon. In order to sample the fluctuations due to surface waves, the path should have surface interactions. The greatest effect on the tomography signal should occur when the ray path is almost perpendicular to the direction of travel of the surface waves[Ref. 9]. As can be seen in Figure 2 on

page 11, lines connecting the receivers to the transmitter would spread over an arc of about 45 degrees from north to northeast relative to the signal transmitter. If the eigenray paths are planar, then these ray paths would be perpendicular to the expected swell direction, that is from the west or northwest.

Theresa Rowan predicted eigenrays and their travel times using the program MPP[Ref. 6]. All of the raypaths had many surface interactions as a consequence of the shallow location of the receivers. It should also be noted that moving the location of the receiver a few meters would give much the same path in deep water but could significantly change the number of surface interactions in shallow water.

Internal waves will also have the greatest effect if propagating perpendicular to the path of the ray. The direction of propagation of internal waves in Monterey Bay is unknown but is expected to vary with orientation with the submarine canyon rim. Internal tidal bores occur in submarine canyons and have been observed in Monterey Bay [Ref. 10]. Internal tides force cold, dense water over the rim of the canyon[Ref. 11]. This may be one of the forcing functions generating the internal waves.

All the receivers were positioned in about 100 meters of water. This was predicted to give several eigenrays without too many bottom interactions (<10 in most cases) which could seriously attenuate the signal[Ref. 6]. This depth still supports the approximations used in ray theory propagation and sea surface waves are only beginning to feel the effects of the continental shelf on their motion[Ref. 9]. The receiver locations will be designated by letters as are shown in Figure 2 on page 11. Stations J, L, L-1, and L-2 are located so that

eigenrays will have relatively little travel in the submarine canyon. Their paths cross the canyon and continue up a fairly gradual slope. Stations G, H, and I require the eigenrays to pass though the most complex bathymetry along the length of the canyon, which could lead to complicated arrival time fluctuations. Station E has a path requiring propagation trough 100 to 200 meter water for almost 20 kilometers. This leads to many surface and bottom interactions which could make the signal too weak to be received. Position B is the closest station and paths to B cross the Carmel Canyon but not the Monterey Canyon.

C. EQUIPMENT

1. Transmitter

The tomography transmitter is a 224 Hz resonant system controlled by a microprocessor and powered by batteries. It was modeled after neutrally buoyant SOFAR floats and is ruggedly designed for deep water use. As shown in Figure 3, it consists of four quarter-wavelength aluminum pipes, each about two meters long, and each driven at the closed end by a piezoelectric driver. The system has a high Q , limiting the useable signal bandwidth to 16 Hz when demodulated to baseband. The central tube contains the batteries, microprocessor, digital to analog converter, amplifier, and clock. The clock is a low-power quartz clock carefully calibrated with respect to temperature for very accurate time keeping. The source is held tightly between a large anchor and glass flotation balls. A tension of about 2,000 pounds with only a 1 meter distance from the anchor is expected to keep transmitter motion to a minimum. Other experiments with long mooring distance have measured the position of the transmitter as it moves in the current [Ref. 2]. For

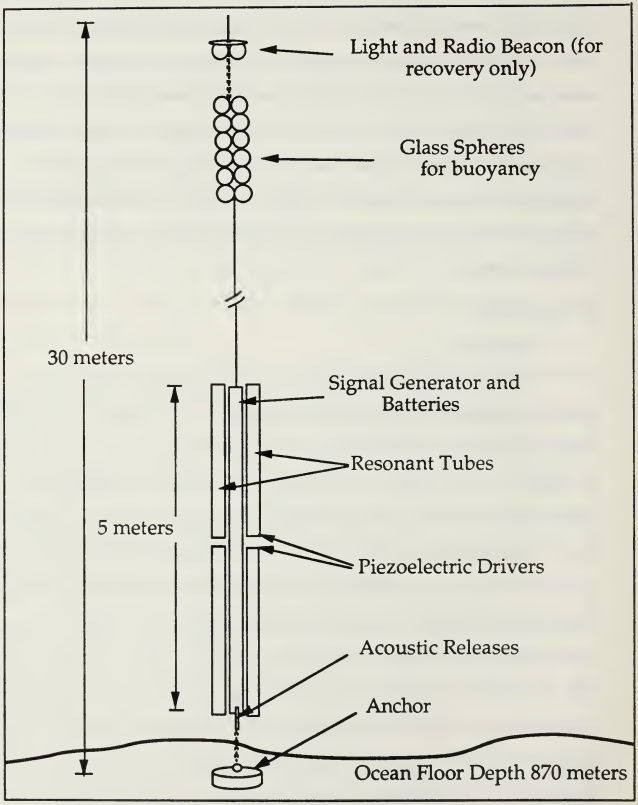


Figure 3: 224 Hz Resonant Tomography Signal Transmitter.

recovery, two acoustically triggered releases are attached to a chain led through the eye on the anchor. Only one release need operate for recovery. The transmitter used in this experiment is one of those used in a 1981 experiment off Bermuda and in several other experiments. It transmitted a phase-modulated signal continuously for four days at an approximate source level of 172 dB re 1 μPa at 1 meter. This same source has been used for intermittent transmission of signals at up to 185 dB re 1 μPa at 1 meter. [Refs. 1,2]

2. Receivers

The acoustic receivers used in the experiment were modified AN/SSQ-57 sonobuoys configured as shown in Figure 4. The unmodified sonobuoys have a single omnidirectional hydrophone connected by wire to a VHF radio transmitter, all powered by a salt-water battery and having a lifetime of about 8 hours. The buoys as modified used the same hydrophone, radio transmitter and antenna but had a longer life battery and an anchor so that they could be used for a longer period. During modification, the antenna and the buoy electronics package were attached to a building foam insulation and plywood float which also supported a waterproof battery compartment. Panasonic LCL12V38P wheelchair batteries were used to power the buoy. The battery could power the buoy for up to one week. The buoys were moored using 15 pound mushroom anchors and about 250 meters of polypropylene line. The hydrophone wire was attached to the anchor line so that the hydrophone would rest on the bottom near the anchor. The sonobuoy electronics packages were modified by Sparton Electronics (manufacturer of

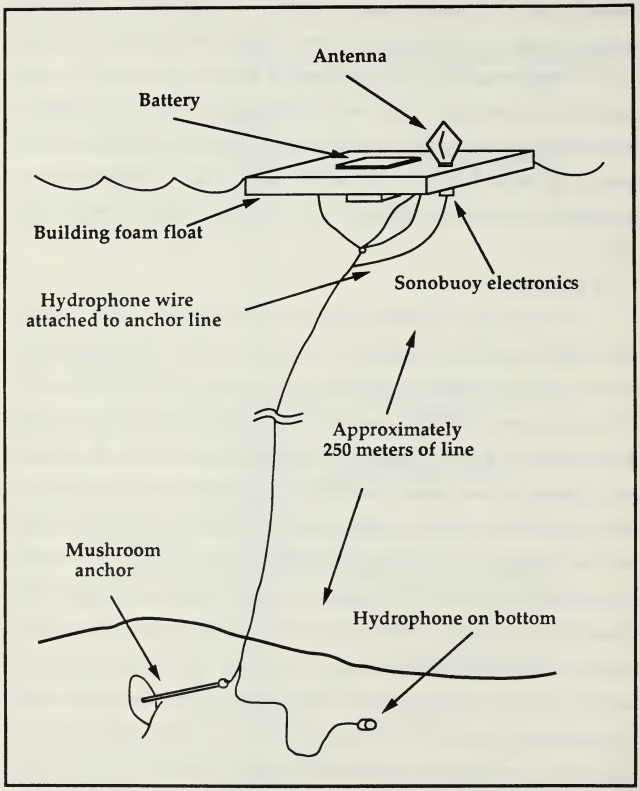


Figure 4: Modified AN/SSQ-57 sonobuoy as used in the Monterey Bay Experiment. The hydrophone rests on the bottom to reduce the possibility of motion.

the unmodified buoys) and installed in the floats and anchor systems by Woods Hole Oceanographic Institution personnel.

A total of 11 modified AN/SSQ-57 buoys were prepared, of which there were several failures. In addition to these, two experimental Moored Inshore Undersea Warfare (MIUW) buoys, AN/SSQ-58, were deployed. One MIUW buoy was deployed with a modified AN/SSQ-57 buoy at station B and the other was deployed alone at station L-1. The data recorded from the MIUW buoys is probably not useable for tomography inversions as the hydrophone is suspended in the water below the floating buoy. Shifts in the travel time of signals received due to buoy motion probably cannot be sorted out from arrivals due to ocean path fluctuations.

The modified AN/SSQ-57 buoys have an acoustic bandwidth from 10 Hz to 20 kHz. The AN/SSQ-58 MIUW buoys have a useable acoustic bandwidth from 50 Hz to 10 kHz. Both types use an FM radio transmitter with a transmitted power out of about .5 to 1 watt on any of 31 selectable VHF channels.[Refs. 12,13]

3. Acoustic Data Recording

The sonobuoys transmit to a receiver in a van located on Huckleberry Hill during the experiment. Huckleberry Hill on the grounds of the Defense Language Institute (DLI) at the Presidio of Monterey is one of the highest unobstructed points on Monterey Peninsula. The antenna on the van is about 260 meters above sea level and can receive VHF and UHF radio signals from Monterey Bay and beyond to a radius of about 60 kilometers, just over 30 nautical miles. Close along the coast to the south of Point Lobos there are areas where radio shadows exist but at Point Sur good reception begins

about 10 kilometers off the coast. Good radio communications were maintained throughout the area of the experiment.

The sonobuoy receiving system, shown in Figure 5, consists of a directional antenna which feeds the received signal through a filter and preamplifier to an AN/ARR-72 sonobuoy receiver. The AN/ARR-72 is a multichannel sonobuoy receiver used by the U.S. Navy in aircraft. The outputs from the receiver are routed to a patch panel where they can be connected to test equipment (for analyzing the signal as it is received) or to the data recording system.

The recording system uses Yamaha Hi-Fi Stereo videocassette recorders (YV-1000) which have been modified to record two audio channels, two digital pulse-code-modulated (PCM) audio channels, and a time code signal on standard commercial videotapes. In this experiment Maxell XL Hi-Fi 120 videotapes on extended play would record 6 hours of data. One audio channel on each tape recorded a 7168 Hz synchronization square wave signal from a signal generator stabilized by a 1 megahertz rubidium frequency standard. This signal is used for accurate demodulation and sampling of the recorded data. When replayed, the time-code signal displays the hour, minute, and second that the data was recorded. Data was normally recorded on the two PCM channels of each recorder but in a few cases the last audio channel was also used. All channels appear to reproduce the signal adequately, 30 Hz to 20 kHz for the PCM channels, with a slight lowering in frequency. The 7168 Hz recorded signal is shifted to 7160.85 ± 0.05 Hz.

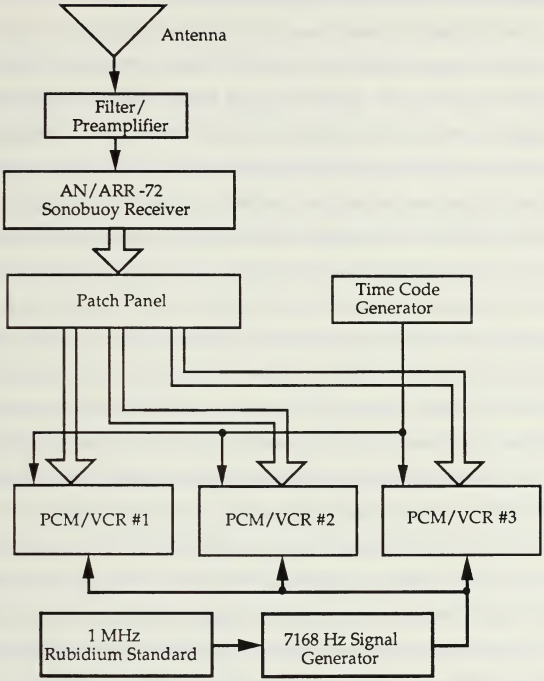


Figure 5: Sonobuoy data recording system located in the van. This system receives the sonobuoy radio transmission, converts it to analog sound, and then records it on videotape using pulse code modulation.

4. NDBC Wave Measurement and ARGOS buoys

The National Data Buoy Center (NDBC) has operated several types of directional wave measurement buoys since 1977. The moored buoys collect surface wave spectrum and direction and are usually equipped with other meteorological sensors such as thermometers and anemometers to help give a complete picture of the weather affecting the sea surface. The buoys measure the surface elevation and wave slope in order to calculate the wave spectrum and direction. The method has undergone extensive testing and has been shown to be accurate in most cases[Ref. 14]. The spectrum coefficients are calculated using a segmented fast fourier transform on 100 seconds of data. An average is made of 19 sequential data segments with an overlap of 49 seconds, giving the data 28 equivalent degrees of freedom. After correcting for various scaling factors resulting from the parabolic windowing used before the transform, the data is ready for transmission. Once an hour the data is transmitted by the buoy to the Geostationary Operational Environmental Satellite (GOES). From the satellite the data is downlinked and relayed to NDBC and other users. The data contains information on wave height and direction as well as the power spectrum from 0.03 to 0.30 Hz with 0.01 Hz resolution. The three meter diameter discus buoy in Monterey Bay (station 46042) also measures wind speed and direction. The buoy is located in deep water (about 2000 meters) southwest of Santa Cruz ($36^{o}45'$N - $122^{o}23.5'$W).

Four free-drifting ARGOS buoys were obtained for additional data collection. Two of the buoys were designed to measure wave spectra in much the same way as the NDBC discus buoy. These are designated TMD by the manufacturer. The other two (designated TZD) suspend a 600 meter

thermistor string below them to make temperature measurements. The buoys were designed and built by Polar Research Laboratory, Incorporated and utilize the ARGOS system for telemetry. ARGOS is a joint program of the CNES (the French space agency), NASA, and NOAA. The ARGOS transmitters are a very simple, small package (<1kg) powered by batteries (approximately 200 milliwatts) and used for many data transmission and tracking systems. The transmitter sends a message of up to 256 bits once every minute at 401.650 MHz automatically, whether there is a satellite overhead or not. Multiplexing occurs at the receiver through random timing of transmissions as well as through doppler frequency shifting due to satellite motion - up to 24 kHz for older TIROS low earth orbit satellites and 80 kHz for newer ones. The location of the transmitter is calculated from doppler shift measurements made by the satellite. Normal accuracy for location is about 300 meters. Typical data delivery time is three hours from uplink. NDBC receives and processes the ARGOS wave buoy data.[Ref. 15]

The ARGOS buoy measurements were expected to supplement the more accurate data from the NDBC moored buoy and from other measurements. The uneven time spacing and random drift pattern of the buoys would decrease the expected usefulness of the data. (Later, a different problem preventing any use of the ARGOS data will be explained.)

5. Sound Speed Profile Measurement

The vertical structure of the sound speed in the ocean determines to a large extent the path sound energy will travel through the ocean. Records of many measurements of the sound speed profile are averaged and kept in databases in order to predict sound propagation through the oceans and this

type of data was used in the initial ray tracing for this experiment. Fluctuations around this average profile are caused by numerous different forces, and to both verify the climatological data and to look for fluctuations the sound speed profiles at various locations were measured. The speed of sound in sea water can be found from an empirically derived function of pressure, salinity, and temperature. The dominant effect in shallow water is the variation of temperature. The salinity of sea water can be calculated from the conductivity of the water and the depth (or density) can be found from the pressure. A set of CTD measurements (conductivity, temperature, density) can be combined to generate a sound speed profile.

In this experiment a digital, recording, battery-powered CTD measuring device manufactured by Neil Brown Instruments was used. This system is powered by a rechargeable battery but is limited by its data storage capacity to about four hours of continuous data collection. After four hours of recording, the CTD data is transferred via cable to a personal computer for storage on floppy disks. In addition to CTD measurements the device measures the transmissivity of light in the water with a low power transmissometer. In use, the CTD device is weighted to help it sink quickly while being lowered by cable from the research vessel. The device could be lowered at about 45 meters per minute and is usually raised at the same rate. A battery powered acoustic transmitter is attached to the frame of the CTD device as a safety precaution. The transmitter "pings" every few seconds and the received sound registers on a recording fathometer trace. As it nears the bottom, the bottom reflected signal grows stronger and also appears on the

trace. The distance between the two signal receptions gives the distance remaining to the bottom and so a collision with the bottom can be avoided.

Several problems are inherent with this device. Because of the limited vertical speed of the device, consecutive measurements in deep water may be separated by intervals of 30 to 45 minutes. Drift of the deploying vessel can easily be greater than one knot (1.8 kilometer/hour) and consecutive measurements might be a kilometer apart. "Yo-yo" measurements, maintaining position as much as possible, may give adequate information about internal wave amplitude but frequency and direction will be difficult to determine.

6. Acoustic Doppler Current Profiler

The acoustic doppler current profiler (ADCP) transmits four narrow, 120 kHz beams of sound in pulses from the bottom of the research vessel. The profiler looks at various time delays of sound scattered back to the transducers to range gate the signal. By measuring the doppler shift of returned pulses in four directions and comparing it to the ship's course and speed from the ship's navigation system, an estimate of the north-south and east-west water velocity as a function of depth is obtained. Because of the boundary conditions of internal wave motion, these create a characteristic movement of the water around the pycnocline related to the horizontal motion caused by vertical displacement in the internal wave. This may appear in the ADCP data.

D. SUMMARY OF THE EXPERIMENTAL PROCEDURE

The Research Vessel Point Sur, operated by Moss Landing Marine Laboratories for the National Science Foundation, was used for deployment

and recovery of all equipment as well as a platform for the CTD and ADCP data measurements. The van located on the hill at DLI began recording when the first sonobuoys were placed in the water. The plan for the experiment was fairly straightforward, but evolved during the experiment as weather, equipment, and luck in locating deployed equipment began to affect schedules. The actual chronology of events is given in Appendix A. The R/V Point Sur was to begin by proceeding south to the seamount and deploying the tomography signal transmitter. During the transit to the north rim of the canyon the four drifting ARGOS buoys would be deployed. Upon reaching the continental shelf at the edge of the submarine canyon, the modified sonobuoys would be deployed, working from west to east. CTD measurements were to be made at each station and, after completion of buoy deployment, the vessel would proceed to different parts of the experiment area to make CTD "yo-yo" measurements. A "yo-yo" measurement is repeated raising and lowering of the CTD to resample the same water column, hopefully to gain information about internal waves at that position. At the end of the experiment, 96 hours after the beginning, the equipment would be recovered, probably in reverse order to the way it was deployed.

III. SIGNAL PROCESSING

A. THEORY OF SIGNAL PROPAGATION

Treating the ocean medium as a large, time-varying distortionless filter, the impulse response of the source-receiver channel is just the sum of the impulse responses of the individual paths

$$h(t) = \sum_{i=1}^{P} a_i \delta(t-\tau_i) \ , \qquad (3.1)$$

where P is the number of paths, a_i is the amplitude, and τ_i is the total travel time along the path. If the transmitted signal is an impulse then the received signal will be the impulse response. The separate paths can be predicted from ray theory.

Ray theory is the result of an approximate solution to the linearized, lossless wave equation[Ref. 16]. In a motionless medium, the equation is

$$\nabla^2 p = \frac{1}{c^2} \frac{\partial^2 p}{\partial t^2} \ , \qquad (3.2)$$

for sound speed c(x,y,z) and pressure p(x,y,z). The wave equation in a motionless medium is a good approximation. Variations in the travel time due to current are generally an order of magnitude less than the changes due to changes in the sound speed and so can be ignored in a first order

treatment.[Ref. 1] For a mesoscale eddy, the currents of 10 cm/s or so at the surface typically diminish to 3 cm/s below the thermocline. For a velocity of 5 cm/s at a depth of 1 km, the corresponding temperature anomaly is approximately 1 degree Celsius. This temperature perturbation results in a sound speed perturbation of approximately 5 m/s, 100 times the change due to current. Also note that a change in the path length (as occurs with surface waves) can have an even greater effect, depending on the size of the change.[Ref. 4]

A solution to the wave equation is:

$$p(x,y,z,t) = A(x,y,z)e^{j\omega[t - \Gamma(x,y,z)/c_0]} , \qquad (3.3)$$

with A the pressure amplitude, c_0 a constant phase speed, and Γ having units of length. Finding surfaces (x,y,z) such that Γ is constant gives surfaces of constant phase. Also, $\nabla\Gamma$ is always perpendicular to the surface of constant phase and pointing in the direction of travel. Substituting into the wave equation gives:

$$\frac{\nabla^2 A}{A} - \left(\frac{\omega}{c_0}\right)^2 \nabla\Gamma\bullet\nabla\Gamma + \left(\frac{\omega}{c}\right)^2 - j\frac{\omega}{c_0}\left(2\frac{\nabla A}{A}\bullet\nabla\Gamma + \nabla^2\Gamma\right) = 0 \ . \qquad (3.4)$$

If A and $\nabla\Gamma$ are limited so that

$$\left|\frac{\nabla^2 A}{A}\right| << \left(\frac{\omega}{c}\right)^2 , \ \left|\nabla^2\Gamma\right| << \frac{\omega}{c} , \text{and} \left|\frac{\nabla^2 A}{A}\bullet\nabla\Gamma\right| << \frac{\omega}{c} , \qquad (3.5)$$

then the equation can be approximated by

$$\nabla\Gamma \bullet \nabla\Gamma = n^2 \qquad \text{with } n(x,y,z) = \frac{c_o}{c(x,y,z)} \quad , \tag{3.6}$$

where n(x,y,z) is the refractive index as a function of location and c_o is an arbitrary reference sound speed. This equation is known as the Eikonal equation. The limits placed on the sound speed structure can be described as:[Ref. 16]

1. The amplitude of the wave must not change appreciably in distances comparable to a wavelength.
2. The speed of sound must not change appreciably in distances comparable to a wavelength.
3. The channel depth and source-receiver distance must be large in comparison to a wavelength.

If these conditions cannot be met, other methods must be used, and "full wave" or modal solutions can be attempted[Refs. 4,16]. Only ray solutions will be discussed in this thesis.

Since $\nabla\Gamma$ at each position defines the direction of the wave's motion, the point by point solution gives the path traversed by the acoustic energy. The travel time for a ray path can be found by integrating the sound slowness (inverse speed) over the specific ray path denoted by P_i (the ith ray path)

$$\tau_i = \int_{P_i} \frac{ds}{c(x,y,z,t)} \quad . \tag{3.7}$$

The fluctuations in sound speed can be thought of as perturbations from some arbitrary base speed $c_o(z)$,

$$c(x,y,z,t) = c_o(z) + \delta c(x,y,z,t), \tag{3.8}$$

so that the travel time becomes a constant travel time with a perturbation

$$\tau_{i,o} + \delta\tau_i = \int_{P_i} \frac{ds}{c_o(z) + \delta c(x,y,z,t)} \, . \tag{3.9}$$

For $\delta c << c_o$, an approximation from the binomial expansion can be used

$$\begin{aligned} \tau_{i,o} + \delta\tau_i &= \int_{P_i} \frac{1}{c_o(z)} \frac{ds}{\left(1 + \dfrac{\delta c(x,y,z,t)}{c_o(z)}\right)} \\ &\cong \int_{P_i} \frac{1}{c_o(z)} \left(1 - \frac{\delta c}{c_o(z)}\right) ds \\ &\cong \int_{P_i} \left(\frac{1}{c_o(z)} - \frac{\delta c}{[c_o(z)]^2}\right) ds \, . \end{aligned} \tag{3.10}$$

The perturbation is

$$\delta\tau_i = -\int_{P_i} \frac{\delta c(x,y,z,t)}{[c_o(z)]^2} \, ds \, . \tag{3.11}$$

The inverse problem is to determine $\delta c(x,y,z,t)$ from $\delta\tau_i$. The travel time perturbation $\delta\tau_i$ depends on the magnitude of sound speed fluctuations and the path of the ray, which determines the water that is sampled by that ray. Note that this perturbation relation has now been linearized. Inverse mathematical methods are often used in connection with geophysical problems where some characteristic is measured by its effect in perturbing some transmitted signal, rather than direct observation of that characteristic. There is a large body of information relating to linear and nonlinear inverse techniques - many of which can be applied to acoustic tomography inversions. [Ref. 17]

The solution of the the ocean acoustic tomography problem is tied directly to the "forward" problem. The path of each eigenray between the source and receiver must be identified before the integral relating time perturbation to sound speed perturbation can be inverted. This eigenray is normally considered to be fixed spatially (usually a good approximation) with the sound speed perturbations acting on this path. Fluctuation in the sound speed field is the data upon which ocean acoustic tomography depends, but if the fluctuation is too great, the ray path may become unstable and no longer reach the receiver. Rays do not arrive as a single point but cover an area measured by the Fresnel zone size. The size of the Fresnel zone depends on the sound speed structure and acoustic frequency but for channel transmission remains fairly constant after 20 kilometers.[Ref. 4] This size and knowledge of sound speed fluctuations along the path can be used to estimate path stability.

In summary, ocean acoustic tomography requires a sufficient understanding of the ocean along the source-receiver path that eigenrays along which the signal will travel can be predicted. The received signal must have an "arrival" structure which is stable and does not fade or disappear. The arrival must be identifiable as to its path for the tomographic inversion to proceed. The transmitted signal must be constructed to facilitate an accurate estimate of the travel time perturbations and should be resolvable form other arrivals at very close intervals. Finally, these time perturbations will be used to estimate the fluctuations in the ocean sound speed field using inverse methods.

B. SIGNAL DESIGN

1. System Requirements

The basic task of signal processing in tomography is to receive the tomographic signal, decompose the received signal into individual eigenray arrivals, and estimate the arrival time of these arrivals. The processing should assist in improving the signal-to-noise ratio of the received signal if this can be done without an adverse effect on the received data. Eventually, the signal-to-noise ratio will limit the accuracy of the estimation of the arrival time. This chapter will discuss the various questions involved in signal design and processing and the solutions chosen in this experiment.

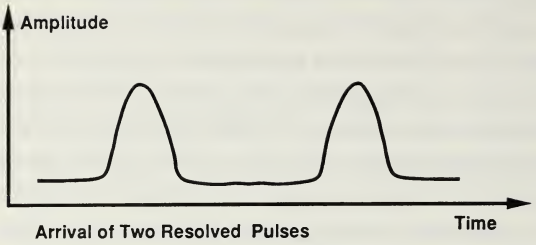
2. Signal Resolution

The time separation required of signals traveling along different eigenrays can be predicted by ray tracing programs such as MPP. The results determined by Theresa Rowan give predictions of arrival time separations between consecutive ray arrivals ranging from 2 to 500 milliseconds, with

most separated by more than 80 milliseconds. In order to separate the closest arrivals, the signal would have to be less than 2 milliseconds in duration. The tomographic source that was used in the Monterey Bay experiment has a bandwidth of only 16 Hz, limiting the shortest pulse that can be efficiently transmitted to about 62.5 milliseconds (1/16 Hz). Pulses arriving with less than 62.5 milliseconds separation may appear as one pulse of greater amplitude, and not be resolvable into separate pulses. Figure 6 shows an example of such an arrival. In this experiment, the transmitter bandwidth of 16 Hz limited the signal to a minimum period of 62.5 milliseconds, even though that could not resolve all eigenrays into distinct arrivals.[Ref. 6]

3. Pulse Compression

A finite length pulse signal is the closest practical equivalent of an impulse (a signal of infinitesimal length and infinite magnitude) that can be transmitted. The amplitude and length of the pulse are limited by the peak power and bandwidth, respectively, of the transmitter. An effective method of boosting the peak amplitude is pulse compression. Pulse compression has been used extensively in RADAR applications but to a lesser extent in underwater sound transmission[Ref. 18]. Simply stated, a long coded signal is transmitted and the received signal is passed through a matched filter which compresses the long transmission into a short, high energy pulse. One relatively easy technique for doing this is to use maximal-length sequences. This method uses a phase-modulated carrier signal to transmit a specific maximal-length code. The autocorrelation of the code with the received signal produces a single pulse at the point where the code and received signal match with an increase in amplitude equal to the number of digits in the



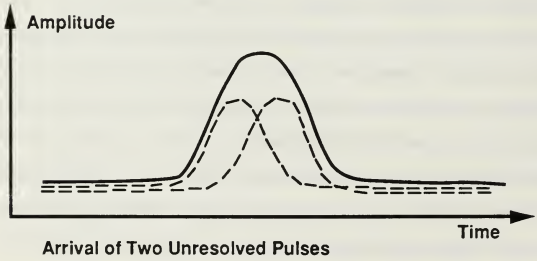


Figure 6: Comparison of Resolved and Unresolved Pulses. If two pulses arrive without enough time separation they will overlap and appear as a single long duration pulse.

code. The width of the pulse is equal to the width of one individual digit of the code. The length of the code is only limited by the system limitations for which it will be applied so the amplitude gain can be quite large when compared to the power the transmitter can send in a single pulse. Appendix B discusses the generation and autocorrelation of the maximal-length sequences.

4. Signal Period

The maximal-length sequence consists of a number of digits determined by the order of the sequence. The code is transmitted continuously, phase modulating a carrier frequency, for the period of the sequence. If the code is transmitted at the maximum rate allowed by the bandwidth of the transmitter, the length will be determined as a compromise between two characteristics:

1. The shorter the code length, the greater the repetition frequency and the higher the sampling frequency for ocean data. This determines the highest frequency which may be observed.
2. The longer the code, the greater the increase in signal-to-noise ratio of the signal and the more accurately the arrival time of the signal can be estimated.

The driving consideration in this experiment is the period of the surface waves to be investigated - fully developed seas of greater than 5 seconds period. To sample the fluctuations due to the surface waves at the Nyquist frequency, the period of the signal must be less than 2.5 seconds. A maximal-length sequence 31 digits long transmitted at a digit frequency of 16 Hz has a period of 1.9375 seconds. This length was chosen for the Monterey Bay

experiment. As discussed in Appendix B, the code is generated from a primitive polynomial. The polynomial for this case is

$$g(D) = D^5 + D^2 + 1 , \qquad (3.12)$$

resulting in the (reverse) code

$$M^T = [0\,0\,0\,0\,1 \;\; 0\,0\,1\,0\,1 \;\; 1\,0\,0\,1\,1 \;\; 1\,1\,1\,0\,0 \;\; 0\,1\,1\,0\,1 \;\; 1\,1\,0\,1\,0 \;\; 1] . \qquad (3.13)$$

This code is mapped from {0,1} to {1,-1} and used to phase modulate a 224 Hz carrier signal. The transmitted signal is given by

$$s(t) = \cos (2\pi f_c t + M_i \theta) , \qquad (3.14)$$

where f_c = 224 Hz and M_i is the ith digit in the (mapped) maximal-length sequence. The power spectrum of this signal has an envelope characterized by the familiar (sin(x)/x) squared function

$$P(f) = \left(\frac{\sin (\pi df)}{(\pi df)} \right)^2 , \qquad (3.15)$$

where d is the digit period. The envelope is filled by impulse functions separated by the code repetition frequency. Some advantage can be taken of this. If θ is chosen so that

$$\theta = \tan^{-1}(\sqrt{N}) \qquad (3.16)$$

for N equal to the number of digits in the code, the carrier signal will fall exactly on the envelope and result in the maximum signal-to-noise performance after demodulation and pulse compression.[Ref. 18]

5. Arrival Time Estimation

The resulting pulse after pulse compression of the maximal-length sequence is a flat topped pulse of one code digit duration. The estimation of the arrival time of the pulse must produce two results:

1. Find a characteristic of the received pulse which can be reliably located on each arrival and the arrival time estimated.
2. Estimate the uncertainty in the arrival time estimate.

Because the signal is transmitted with a finite bandwidth and suffers some dispersion during its travel, the received signal is rounded at the edges of the pulse, sometimes so much that it resembles the peak of a Gaussian distribution curve. One method of finding a consistent point on each pulse arrival is to correlate the signal again with a square pulse of the same duration as the signal. For the perfect received flat topped pulse, this is the correlation of two squares, the result is a triangle with the peak at the center of the signal. In effect, this gives the received signal a sharper peak. Since this processing is done using discrete points of much greater separation than the expected uncertainty in the best estimation, the position of the peak is found by interpolation. Various methods such as parabolic fit, Gaussian fit, and cubic spline are available to fit curves to the discrete points with a separation of points somewhat less than the expected uncertainty. The time of arrival of

the interpolated (or original) point with the highest magnitude is the arrival time estimate. If the code is transmitted continuously then the arrival time is compared to an arbitrary starting point recurring at the code repetition frequency.

The accuracy of the time estimate depends on the bandwidth of the signal and the received signal-to-noise ratio. The calculation of this type of non-linear estimate with white Gaussian noise is discussed by Van Trees[Ref. 19]. Spindel gives the result as

$$\sigma_t = \frac{1}{2\pi B \sqrt{SNR}} \quad , \tag{3.17}$$

with σ_t the arrival time uncertainty, B the bandwidth of the signal, and SNR the signal-to-noise ratio.[Ref. 18] For 10 dB signal-to-noise ratio and a 16 Hz bandwidth, this equation gives an uncertainty σ_t of 3.1 milliseconds.

C. SIGNAL DEMODULATION AND CORRELATION SYSTEM

1. Analog Processing

The received acoustic signals from the sonobuoys are recorded on videotape for storage. This analog or pulse-code-modulated recording is played back for quadrature demodulation and digitization as shown in Figure 7. The tomography signal is contained in a band 16 Hz above and below the carrier frequency of 224 Hz. In order to ensure the proper frequencies and timing of the tape recordings, the 7168 Hz synchronization signal is used both

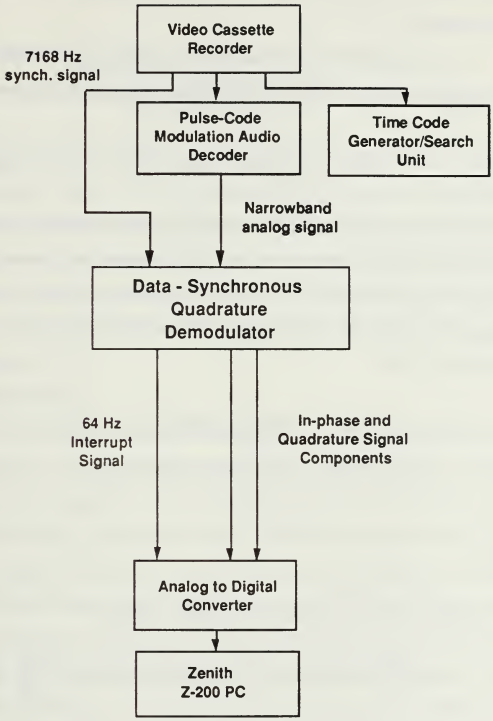


Figure 7 : Quadrature Demodulation and Digitization .

to demodulate the signal and to generate an interrupt signal for the analog to digital converter.

An unsynchronized demodulation system must demodulate the signal without knowing its phase. The received signal can be represented as

$$s(t) = A \cos\left(2\pi f_c t + M_i\theta + \phi\right), \tag{3.18}$$

which is the same as the transmitted signal but with an unknown phase shift ϕ caused by the delay due to the travel time. Because this phase is unknown, the signal must be multiplied by both the cosine and sine at the carrier frequency to recover all of the magnitude of the signal in the baseband. Multiplication gives

$$\begin{aligned} I(t) &= A \cos\left(2\pi f_c t + M_i\theta + \phi\right) \cos\left(2\pi f_c t\right) \\ &= \frac{A}{2}\left[\cos\left(M_i\theta + \phi\right) + \cos\left(4\pi f_c t + M_i\theta + \phi\right)\right], \end{aligned} \tag{3.19}$$

and

$$\begin{aligned} Q(t) &= A \cos\left(2\pi f_c t + M_i\theta + \phi\right) \cos\left(2\pi f_c t + \frac{\pi}{2}\right) \\ &= \frac{A}{2}\left[\cos\left(M_i\theta + \phi - \frac{\pi}{2}\right) + \cos\left(4\pi f_c t + M_i\theta + \phi + \frac{\pi}{2}\right)\right]. \end{aligned} \tag{3.20}$$

These signals are passed through a low-pass filter to remove their high frequency components and produce the in-phase and quadrature signals

$$I_{LP}(t) = \frac{A}{2}\cos\left(M_i\theta + \phi\right) \tag{3.21}$$

and

$$Q_{LP}(t) = \frac{A}{2}\cos\left(M_i\theta + \phi - \frac{\pi}{2}\right)$$
$$= \frac{A}{2}\sin\left(M_i\theta + \phi\right). \tag{3.22}$$

These signals are now baseband and limited by both the input bandpass and output lowpass filters to 16 Hz bandwidth. Since all the information is contained at frequencies below 16 Hz, the digital sampling rate for the signal must be greater than the Nyquist frequency of 32 Hz to avoid aliasing. The sample rate chosen for the experiment was 64 Hz. This gives a period between samples of 15.625 milliseconds, or four samples for each digit in the maximal-length sequence.

2. Digital System

The conversion from analog to digital data was accomplished with a Zenith Z-200 PC (6 MHz, 80286 based machine) equipped with a MetraByte DASH 16F data acquisition and control interface board. The mode in which the DASH 16F was used was to scan 4 channels on receipt of an external interrupt signal and store the 12-bit voltage code in memory via Direct Memory Access (DMA). During DMA the computer Central Processing Unit (CPU) is left free to execute other parts of the program. In this manner the code correlation could be performed in parallel with the analog to digital conversion, resulting in a large processing time savings. A diagram of the operation is shown in Figure 8. The Fast Hadamard Transform described in Appendix B was used to perform the matched-filtering for the code correlation with sufficient efficiency to be run concurrently with the

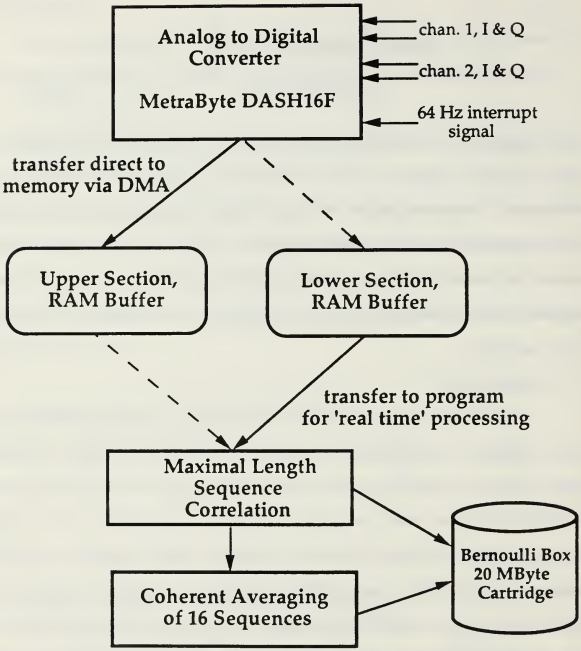


Figure 8: Diagram of tomography signal data flow for 'real time' digitization and code correlation in the program AMORE.

digitization. Equivalent programs performing the correlation using Discrete Fourier Transforms took approximately 300 times as long to perform and could not be used for "real-time" processing of the recorded data.

The in-phase and quadrature components of the signal are combined after the code correlation and are stored as magnitude and phase. This results in about 44 kilobytes of data per channel per minute. This data was stored on 20 megabyte cartridges with a dual drive Bernoulli Box manufactured by IOMEGA. One six hour videotape containing two recorded channels of information was converted to about 17 megabytes of data on each of two cartridges. In addition a coherent average of 16 time periods is conducted and stored. The source code for FORTRAN programs to conduct the signal digitization and correlation are contained in Appendix C. The program AMORE was used for the data conversion with concurrent code correlation and is the program described in this section. The programs AINPUT and AHAD perform the same operations but in two steps, storing the digitized samples before correlating for the maximal length sequence. Both AMORE and AINPUT make use of library routines provided with the DASH 16F board for controlling the board, including the interrupt handler for the external interrupt.

D. TRAVEL TIME ESTIMATION

1. Eigenray Arrival Selection

An important part of understanding the data was an effective display of the data. Programs AGRAF4 and AGRAF5, listed in Appendix C, were used to generate files of magnitude and/or phase for plotting routines in MATLAB and SURFER. MATLAB is a product of The Mathworks, Inc. of Sherborn, MA

and SURFER is a product of Golden Software, Inc. of Golden, CO. Both routines generate a plot usually described as a "waterfall" plot. This plot places one 1.9375 second period of the signal behind another for up to about 70 lines so that any feature common to all the sequences will stand out clearly. For the data coherently averaged for sixteen periods, if every other sequence is skipped, 62 minutes of data can be displayed on a single plot. From these plots an estimate of the resolution and stability can be made by eye. The arrival must not disappear (an indication of an unstable path) and it should not merge or split with another arrival (an indication that the ray arrivals are not resolved).

2. Interpolation between Signal Points

The points of the received signal are separated by the sample period of 15.625 milliseconds. The points can be interpolated to a smaller separation by using curve fitting. A cubic spline curve fitting routine adapted from Press, et al., generates points separated by 0.976 milliseconds[Ref. 20]. Until this point all the calculations have been conducted using integer mathematics. This gives insufficient separation for selecting the highest magnitude point after interpolation. The interpolation is therefore done with floating point decimal mathematics in FORTRAN.

3. Signal-to-Noise Ratio Calculation

Although the point interpolation allows the selection of the time of arrival of the point of highest highest magnitude to less than a millisecond, the actual uncertainty is a function of the signal-to-noise ratio. A pessimistic estimate of the signal-to-noise ratio is made by finding the mean amplitude of all the points in a 1.9375 second data string, not trying to sort out signal from

noise. The peak magnitude is then divided by this value to obtain a signal-to-noise ratio.

4. Methods of Selecting Peak Magnitude

Two different algorithms were used to estimate the arrival time and signal-to-noise ratio. Both programs could perform coherent averaging of consecutive signal periods for up to sixteen periods. This will increase the signal-to-noise ratio but reduces the sampling rate below what is necessary for surface wave data. The method could be of use for investigating internal wave frequency fluctuations. Both programs could also perform a correlation with a square pulse. This correlation results in low-pass filtering of the data and smooths out fast fluctuations(<65 milliseconds) as well as increasing the peak amplitude of features longer than 65 milliseconds. The noise improvement was very slight and the amplitude gain for arrivals did not greatly increase the signal-to-noise ratio or estimation accuracy. The first program, AGONY, is an interactive program which requires the user to input a window size for the program to search for a peak, a starting position, and a minimum threshold for the signal-to-noise ratio. If the maximum amplitude of the peak found does not exceed the SNR threshold, the program stops, displays the signal period in question and asks the operator to pick the peak. The window shifts to take the last peak found as its starting point.

The second program, ACRID, was both less flexible and more efficient. The window for the peak-picking was rigid and the maximum amplitude found inside the window would be the chosen arrival peak. If the signal-to-noise threshold was not attained, the previous arrival time would be repeated with the lower signal-to-noise ratio. The SNR would serve as a

flag for a repeated arrival but would still allow for relatively little noise contamination. Having a uniform separation of the samples is important for Fast Fourier Transform analysis and segmented sample power spectrum estimation. Typical window sizes were about 80 milliseconds to either side of a starting position.

E. SUMMARY OF SIGNAL PROCESSING

The signal processing system estimates the arrival time perturbations from the analog recordings through the following procedure:

1. The signal passes through a band-pass filter to remove any out-of-band noise.
2. The signal is quadrature-demodulated to baseband and low-pass filtered to remove the high frequency components.
3. The signal in-phase and quadrature components are sampled at 64 Hz and digitized.
4. The Fast Hadamard Transform is used to matched-filter for the maximal-length sequence code and the result is stored.
5. Given a certain window around an eigenray arrival, the arrival time of the ray is estimated with respect to an arbitrary code starting position, and the signal-to-noise ratio is calculated.
6. The geophysical time (clock time) of the data point, time of arrival, peak magnitude, and signal-to-noise ratio are stored.

This stored data contains the fluctuations due to path length and sound speed perturbations and will be the input data for the tomographic inversion to estimate the ocean conditions.

IV. PRELIMINARY EXPERIMENTAL RESULTS

A. GENERAL SUMMARY OF THE DATA

1. Acoustic Data

Approximately 300 hours of acoustic data was recorded on videotapes. This data varied because of the location of the receivers and inconsistencies in the operation of the equipment. Ambient noise at all stations was often stronger than the 224 Hz signal but, after the maximal-length sequence correlation, all sonobuoys which functioned showed some ray arrival signature. The amplitude of the received signal varied with time but does not appear to correlate with tidal fluctuations. Interfering acoustic sources were dolphins, whales, and fishing boats. Of these, only the fishing boats adversely affected the signal reception. Radio frequency interference occurred to a greater degree than expected, with most channels having some minor interference and a few having the sonobuoy signal completely blocked for several minutes. Some of the identified sources of interference were a pocket-pager transmitting station, walkie-talkies used by personnel at the Defense Language Institute, marine-band radios, vehicle dispatch radios, and the McDonald's Restaurant radio-intercom for their drive-up window. Most of the interference only degraded the signal for short periods and only on a few channels.

The following is a short description by station of the received data from the time the transmitter was activated:

Station B - 1710 12DEC to 0250 13DEC This is the shortest path and has several resolved arrivals. The buoy appears to have broken free or

been dragged away in the early morning of the second day. The signal-to-noise ratio until then was good.

Station B-1 (MIUW) - 1700 12DEC to 0130 16DEC Several resolved arrivals are present. The arrival structure appears stable but moves quickly, apparently due to buoy motion. The fluctuation in arrival time due to buoy motion probably cannot be sorted out of the motion due to path and sound speed fluctuations.

Station E - 1430 13DEC to 2400 15DEC Only very unstable arrivals with a low signal-to-noise ratio are present. This path travels through shallow water for longer than any other path and bottom losses may have reduced the signal below a useable level.

Station G - 1300 14DEC to 2400 15DEC This stations early data was lost because of a malfunctioning receiver. The data shows one fairly stable arrival and several unstable arrivals. This ray path travels through most of the Monterey Canyon.

Station H - 1330 13DEC to 2230 15DEC Several unstable arrivals are present, usually with a low signal-to-noise ratio. Two dimensional ray tracing may be inadequate to predict the paths of eigenrays which reach this buoy at the head of Soquel Canyon.

Station I - 1300 13DEC to 2200 15DEC Usually several arrivals with good signal-to-noise ratio are present. The arrivals have large magnitude fluctuations and the paths seem to be unstable over a period of hours. Again, the complex bathymetry may lead to unstable three-dimensional raypaths.

Station J - 1300 14DEC to 1400 15DEC This ray path has simpler bathymetry than paths to G, H, and I. After the path crosses the canyon the path has a steady grade into shallow water. Several resolved rays with good signal-to-noise ratio are present. This data record is short because the first sonobuoy at this position never functioned and the second failed after 25 hours.

Station L - 1000 13DEC to 2000 14DEC Many arrivals with good signal-to-noise ratio are present but some of the strongest are sometimes unresolved. This path also had simple bathymetry with a steady slope into shallow water after crossing the canyon. Poor reproduction from a faulty PCM encoder may have contributed to loss of signal at some points. Back up audio tapes will be examined to see if the recording is better.

Station L-1 (MIUW) - 1400 14DEC to 1900 15DEC This buoy required the replacement of a circuit board before it could be deployed. The ray arrivals varied from two resolved arrivals to many unresolved arrivals. The shifts due to buoy motion are not as apparent as for station B-1.

Station L-2 This buoy failed and was immediately recovered.

Data for Station J will be presented as an example data set.

2. Surface Wave Data

The NDBC moored surface wave buoy operated as designed and hourly reports of the surface wave power spectral density, wave direction, barometric pressure, and temperature for the entire experiment have been received. This data will be compared to the data derived from tomography in the same frequency band. Unfortunately, all the data from the ARGOS drifting buoys is unusable. The algorithm used in calculating the power spectral density from the accelerometer inputs uses the lowest frequency information (.01 and .02 Hz) to calculate a noise correction factor. Somewhere in the process an error was made and since neither the raw data nor the correction factor is transmitted or recorded, the correct results cannot be calculated. The data from the moored NDBC buoy should be sufficient.

3. Sound Speed Profile and Current Measurements

The data taken during conductivity, temperature , and density (CTD) measurements and by the acoustic Doppler current profiler is being analyzed at Woods Hole Oceanographic Institution. The sound speed profile results for two positions near the path to Station J will be presented. The ADCP data is not ready at the time of writing.

B. STATION J DATA

1. Station J Eigenray Prediction

The bathymetry along a two dimensional slice between the transmitter and the receiver and the eigenray predicted by Theresa Rowan using MPP are shown in Figure 9 [Ref. 6]. Although the eigenray was predicted from a historical sound speed data base, the measured profile in deep water very nearly matched and the original prediction is probably accurate enough until a three dimensional prediction can be made. The single eigenray predicted has few interactions with the surface or bottom before reaching the the shelf water. Once in the shallow water of the shelf the ray has many reflections. The number of bounces predicted was seven but a small change in the angle of the ray could easily double or halve the number of surface interactions in the last 8 kilometers before the receiver.

2. Measured Sound Speed Profiles

Sound speed profiles from two positions near the ray path connecting the transmitter and Station J are shown in Figures 10 and 11. Figure 10 shows the profile for shallow water at $36^{o}51.095$N-$122^{o}04.798$W, near Station J. The profile in Figure 11 is from deep water at $36^{o}32.906$N-$122^{o}16.210$W, near the transmitter. Both profiles show two traces, one for

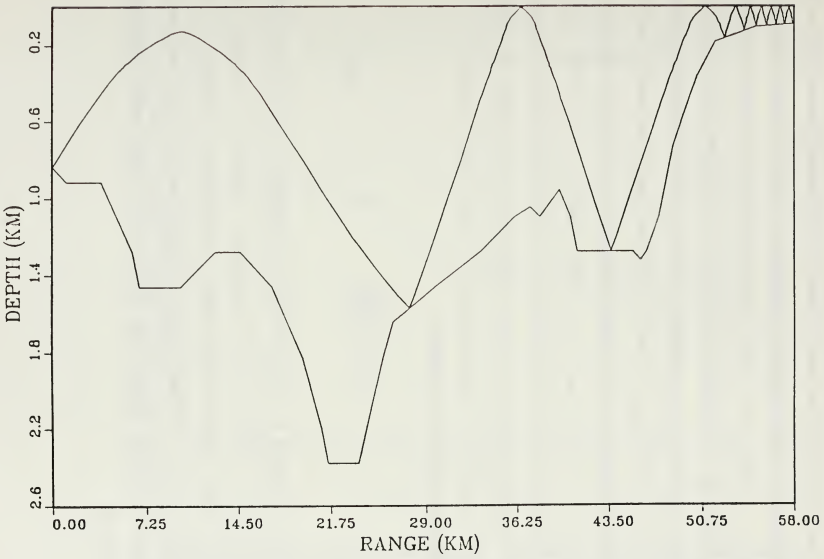


Figure 9: Two dimensional ray path predicted using MPP. This eigenray connects the transmitter at Station A to the receiver at Station J .[Ref. 8]

measurements while the CTD instrument is descending and the other while its ascending through the water column. The difference between the curves where the sound speed gradient is steepest is evidence of internal waves. The difference in depth of a certain sound speed gives a minimum vertical displacement for the internal wave but gives no information about the period or actual amplitude of the oscillation. The two points at 30 meters on Figure 10 are about 4 minutes apart, based on a 30 meter per minute rate for the CTD. Similarly, in Figure 11, the 100 meter points were crossed about an hour apart, based on a 45 meter per minute rate. Analysis of the CTD "yo-yo" measurements may give information on the deep water, lower frequency internal waves however no "fast" measurements were made in shallow water. Using the CTD measurements, the Brunt-Väisälä frequency at the density gradient can be calculated as

$$n = \sqrt{-\frac{g}{\rho}\frac{d\rho}{dz}} \quad , \tag{5.1}$$

for depth z, density ρ, and gravitational acceleration g. The Brunt-Väisälä frequency is the highest where the gradient is greatest. In the case of the shallow water with the sound speed profile shown in Figure 10, the minimum period (maximum frequency) the gradient will sustain is about 8.3 minutes. The density gradient can support much longer period oscillations also. [Ref. 4]

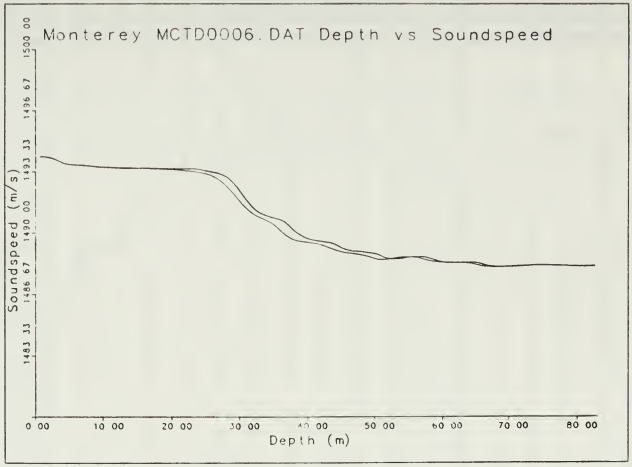


Figure 10: Sound speed profile from position 36°51.095N - 122°04.798W, near Station J. Note that any ray path in this position will be refracted downward. The trace has two lines, one as the CTD goes down and the other as it is brought back to the surface.

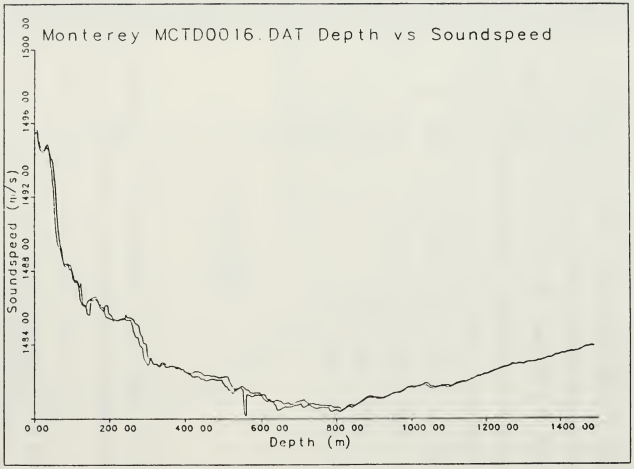


Figure 11: Sound speed profile from position 36°32.906N - 122°16.210W. This profile is typical of the profiles found in deep water at the time of the experiment and very close to the profiles used in MPP for eigenray prediction.

3. Received Acoustic Signal

The data recorded for Station J only covers 25 hours during the experiment because of failures in the first and second modified sonobuoys placed there. The received signal shows three or four ray arrivals throughout the functioning span. All of the arrivals fluctuate in strength over time. Shown in Figure 12 is an example of the received signal. This plot is the result of coherent averaging of 16 sequences and then only plotting every other averaged sequence. The data shown covers 62 minutes for each plot and is used for determining which arrival to track for the travel time fluctuation data. The orientation of the plot assists in visually integrating the data to spot characteristics recurring at the code repetition frequency. The remaining plots for Station J are in Appendix D. Note that the data from different videotapes has a new arbitrary starting point for timing the arrival estimations. This random displacement is unimportant when measuring ocean perturbations with periods somewhat shorter than six hours. If investigation of tidal frequency phenomenon was a goal of this experiment then some method of synchronizing the different data sets would be required. The individual eigenray arrivals can be located (for stable paths) on different tapes by observing the location of a ray relative to the others. The analysis of one ray arrival will be shown to demonstrate the data for travel time fluctuations. In Figure 12, the selected arrival has its peak at about 0.85 seconds after the arbitrary start point as shown on the sequence repetition time scale. While the signal-to-noise ratio of this arrival varies, there is always enough so that it can be measured during the 25 hour interval.

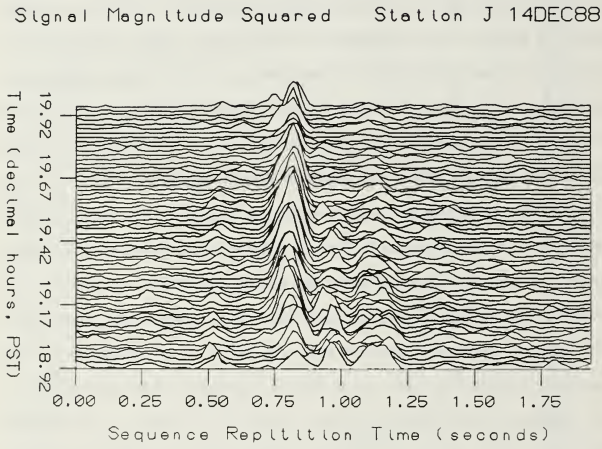


Figure 12: Received acoustic signal after matched-filtering for maximal length sequence from Station J, 14DEC88 1855 to 1957 PST. Each line is 31 seconds of data coherently averaged to one 1.9375 second period. The earliest period is in the foreground and the latest is at the back.

4. Travel Time Fluctuations

The arrival time is estimated by finding the peak of the ray arrival signal. The absolute travel time is something around 50 seconds and is not measured. Each cycle of the maximal-length code is the same as the others and cannot be identified. Moreover, since the fluctuation of the absolute travel time is the same as the fluctuation in the arrival time as measured from an arbitrary starting point, only the latter will be measured. The arrival time estimation vs. time for the selected arrival shown in Figure 12 is shown in Figures 13 and 14. The uncertainty calculated from the signal-to-noise ratio is between 2.5 and 4.5 milliseconds for most of the estimates. The perturbations have a peak-to-peak amplitude of about 50 milliseconds. Also visible are some lower-frequency oscillations.

C. ANALYSIS OF ARRIVAL TIME FLUCTUATIONS AT SURFACE WAVE FREQUENCIES

The power spectral density of the arrival time fluctuations caused by the surface waves should reflect the power spectral density measured by the NDBC surface wave measurement buoy. The power spectral density of the arrival time perturbation was estimated using a segmented Fast Fourier Transform. The individual segments were chosen to be 64 samples long to match the frequency resolution of the NDBC data. Approximately 2.2 hours of data points provides 64 segments for 128 degrees of freedom. An example of the arrival time power spectrum is shown in Figure 15. The resolution bandwidth is 0.00806 Hz. This value is used to normalize the magnitude so that other spectra of the same data will have a directly comparable magnitude although a different resolution bandwidth is used. Note that the segmented

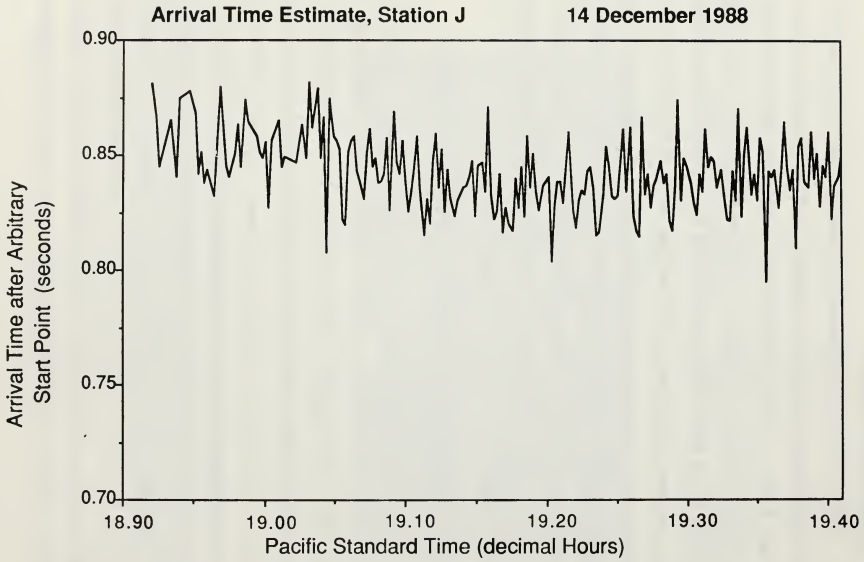


Figure 13: Arrival time estimate for Station J from 1855 to 1924 PST on 14DEC88. The fast fluctuations in arrival time are due to surface waves changing the path length. Lower frequency oscillations from other causes are also seen.

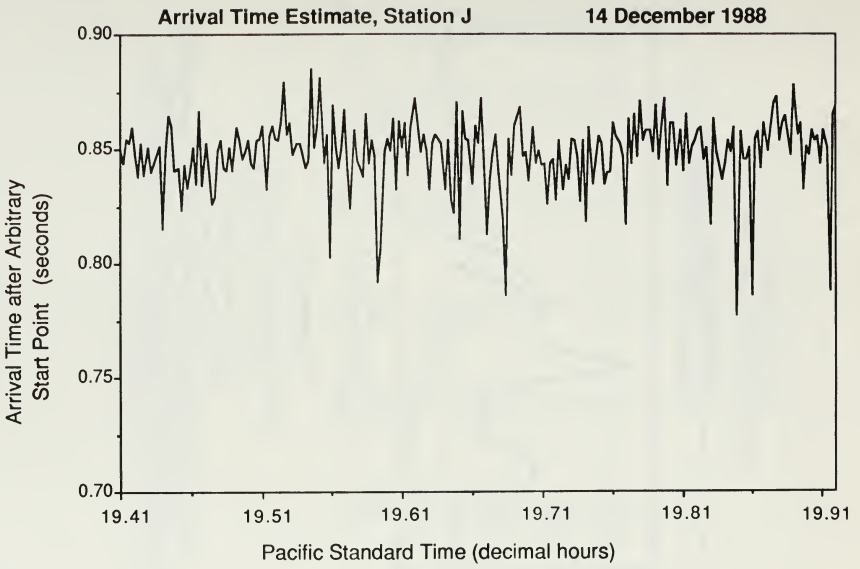


Figure 14: Arrival time estimate for Station J from 1925 to 1955 PST on 14DEC88. The fast fluctuations in arrival time are due to surface waves changing the path length. Lower frequency oscillations from other causes are also seen.

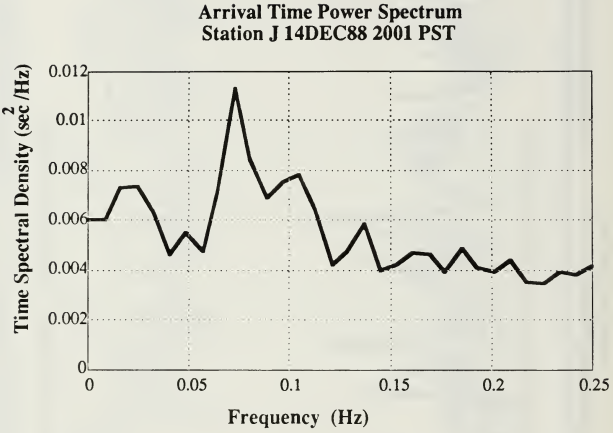
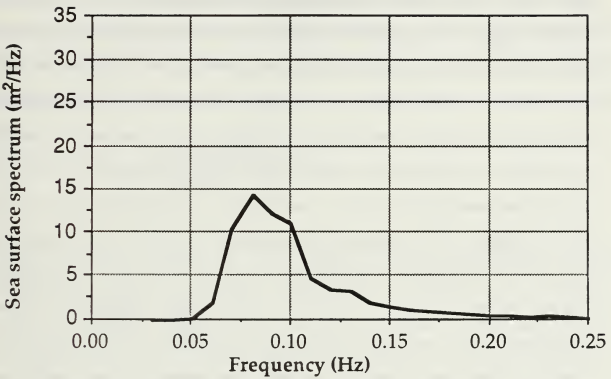


Figure 15: Arrival time power spectrum for Station J. Spectrum from 2.2 hours of Arrival Time Series,1855 to 2107 14DEC88 PST



Significant Wave Height 4.10 m
Average Period 9.67 sec
Dominant Period 12.50 sec
Dominant Direction 308°N

Figure 16: Surface wave power spectrum in Monterey Bay at 2000 PST on 14DEC88 as taken from the NDBC wave measuring buoy southwest of Santa Cruz.

transform method sums (instead of averaging) the result of the FFT's so that the total power will contribute to the magnitude. [Ref. 21]

The power spectrum from surface waves provided by the National Data Buoy Center has already been described. An example of the wave data is shown in Figure 16. The spectral resolution is 0.01 Hz. Additional sea surface and arrival time spectra are included in Appendix D. A comparison of the arrival time and surface wave power spectra immediately shows agreement in the general shape and frequency distribution with the largest concentration of power in the long period swell frequency region of 0.07 to 0.09 Hz. The arrival time spectrum also shows a smaller but still significant peak at about 0.03 Hz. This is a longer period than is normally observed for sea swell in the Pacific. This frequency of fluctuation is higher than can be attributed to internal waves and must be due to a path length change, but either a modulation on the swell or an extremely long period wave could cause it. A possible explanation is "beating" between two systems of long period swell propagating in slightly different directions. A source of this surf beat could be swell that has been reflected or refracted off the shallow water or shoreline along the north side of the Bay.

The arrival time spectrum shows a nearly white noise floor. This is due in part to the random uncertainty in the estimation of the arrival estimation. All fluctuations of higher frequency than 0.258 Hz will spread out the arrival pulse width and lower the signal-to-noise ratio, contributing to the uncertainty. The spectrum for the surface wave buoy data does not show this kind of noise. The algorithm for calculating the wave data calculates a noise correction factor from the two lowest frequency data points, 0.01 and 0.02 Hz,

and applies this to the rest of the data. Since the accelerometer calculations are most sensitive to noise and least sensitive to motion at the lower frequencies this is convenient. Unfortunately, the energy seen by the tomography signal may indicate that "zeroing" the low frequencies may not always be correct.

A relation between the travel time perturbation spectrum for a set of ray arrivals to the frequency-direction spectrum is given by Miller, et al.[Ref. 9] For the case where the wave propagation is perpendicular to ray, the relation is

$$T(\Omega) = \left(\frac{2 \sin \theta}{c_o \Omega} \right)^2 \frac{M 2 \pi g}{\Delta r} F(\Omega,0) \quad , \qquad (5.2)$$

where Ω is the frequency, M is the number of surface reflections, g is the acceleration of gravity, Δr is the ray skip distance, and $F(\Omega,0)$ is the frequency-direction spectrum perpendicular to the ray path. Before this can be calculated, the wave and arrival spectra will have to be normalized and the solution for the number of the surface reflections will have to be found.

D. ANALYSIS OF ARRIVAL TIME FLUCTUATIONS AT INTERNAL WAVE FREQUENCIES

The magnitude of time fluctuations at internal wave frequencies is expected to be of somewhat lower magnitude than fluctuations due to surface waves. Figure 10 shows a sound speed difference of only 5 meters per second across the thermocline, a change of only 0.33%. The surface wave causes a 300 times greater time perturbation for the same amplitude as an internal wave. To begin analysis of the data, the time perturbation data series was detrended by subtracting the mean and then low-pass filtered with an 8th order Chebyshev digital filter to a cutoff frequency of .01 of the original maximum

digital frequency. Oscillations of period greater than 6.4 minutes pass through the filter including any perturbations due to internal waves. The result for Station J is shown in Figures 17, 18, 19, and 20. This data appears to show the presence of several different frequencies but segmented FFT methods were unsuccessful in measuring their distribution. It is probable that the the record length before the oscillations become uncorrelated is not long enough to form a statistically significant group and still have the frequency resolution necessary to analyze the waveform. Other methods such as the Prony method, maximum entropy method, or a frequency-time spectral density may identify these frequencies[Refs. 20,21].

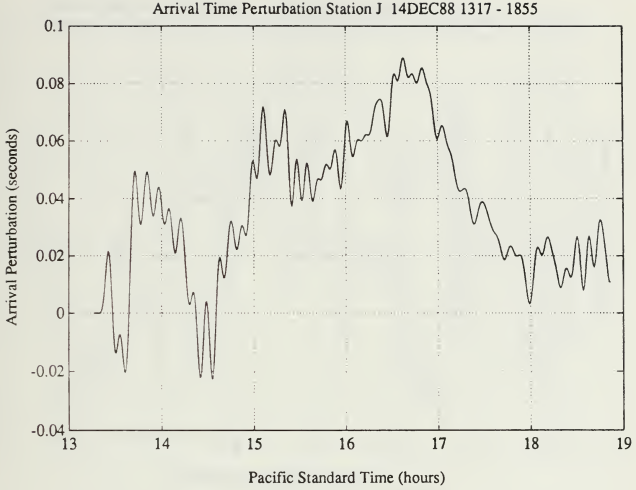


Figure 17: Arrival Time Perturbation Data Low-pass Filtered to .00258 Hz (Periods> 6.4 Minutes)

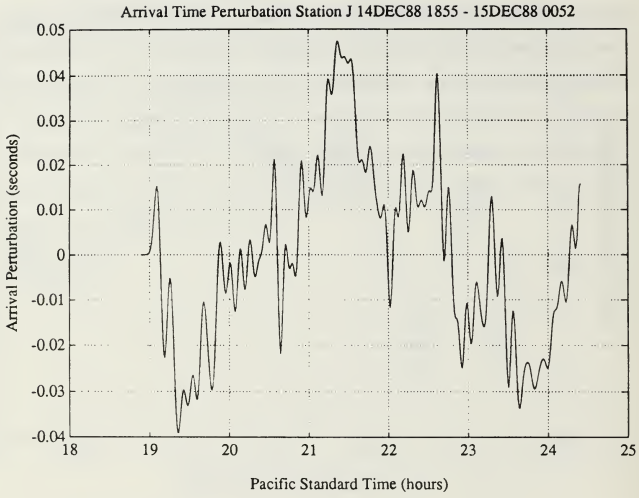


Figure 18: Arrival Time Perturbation Data Low-pass Filtered to .00258 Hz (Periods> 6.4 Minutes)

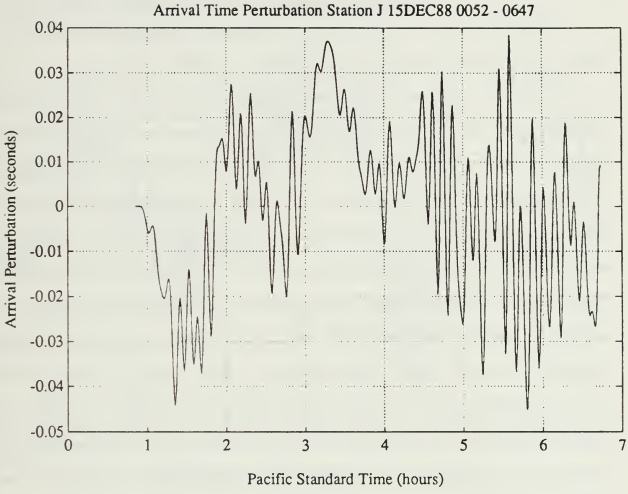


Figure 19: Arrival Time Perturbation Data Low-pass Filtered to .00258 Hz (Periods> 6.4 Minutes). High amplitude after 0400 is due to low SNR during storm

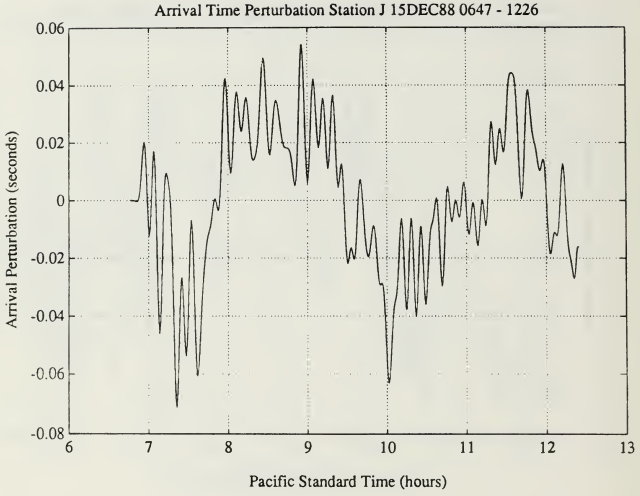


Figure 20: Arrival Time Perturbation Data Low-pass Filtered to .00258 Hz (Periods> 6.4 Minutes)

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

1. Signal Processing Objectives

The system of signal processing using data recorded with a time synchronization signal meets its requirement for accurate timing with an estimated accuracy of under ± 1 millisecond loss over 6 hours, and sufficient signal-to-noise after processing with 7 - 12 dB SNR for most channels. This allows adequate precision in arrival time estimation, 2 - 4 milliseconds, for tomographic analysis. The algorithms developed for use in correlation for the maximal-length sequence are efficient enough to digitize and matched-filter two channels in real-time. The system uses a Zenith Z-200 PC (6MHz, 80286 machine) to store the data on 20 Mbyte IOMEGA Bernoulli Box cartridges, each of which will hold about 6 channel-hours of data.. The method used in the program ACRID for estimating arrival times of the pulse-compressed arrival depends on the operator selecting a window in which the single resolved arrival occurs. The program reliably interpolated to find the peak and recorded the arrival estimate as well as the signal-to-noise ratio, which was used to calculate the uncertainty in the measurement. The accuracy of the system is sufficient for use in acoustic tomography for surface and internal waves. In accomplishing this, the thesis has met its objectives.

2. Tomography Experiment Objectives.

The goals of the tomography experiment have not yet been completely attained. The experiment is a success in demonstrating that the

new tomography system is capable of making the measurements required for the acoustic data with real-time transmission to a shore receiving station. The effects of surface waves and internal waves on the signal are apparent but more work on the path identification and the energy normalization of the power spectra must be done before the tomographic solution can be found. The three-dimensional ray tracing program is not yet ready to predict the more complex paths expected in the Monterey Canyon. In short, the experiment data analysis is not finished, but the preliminary results are very promising.

3. Summary of Results

The results of travel time estimation show that the tomography signal travel time is influenced by both the surface waves and internal waves. The results obtained in the power spectrum of the surface waves and their agreement with the NDBC surface wave buoy measurements is probably the most interesting result of this thesis, as this was the first time such a comparison was made. Another interesting characteristic of the arrival time perturbation power spectra is the peak near .03 Hz. This frequency band is usually not measured in wave buoys but is shown here to be a real and long lasting phenomenon. The internal wave effects are present but to quantify them would require analysis with the "yo-yo" and ADCP data. It is in progress now. The CTD data will be incorporated into the solution of three-dimensional ray paths.

This experiment was innovative in having a fast sampling rate and continuous transmission of the tomographic signal. A major difference from other experiments was the real-time transmission of data to shore, allowing

this massive amount of data from continuous transmission to be efficiently stored. The fact that this experiment was conducted in shallow water with many surface interactions and strong internal waves also changes the magnitude of the perturbations seen. The travel time fluctuations in this 60 kilometer experiment are 5 to 10 times greater than those seen in 300 kilometer experiments with no surface interaction.

B. RECOMMENDATIONS FOR ADDITIONAL WORK

To complete the processing and analysis there are many tasks remaining. The following is a list of some of the topics remaining to be investigated.

1. The arrival time fluctuation estimation must be completed for all the channels. Some arrivals are only stable for a few hours. However, short term analysis on the surface wave field can probably be done with these arrivals.
2. The forward acoustic propagation problem is not completely understood. Three-dimensional ray tracing needs to be done to estimate the ray paths more accurately.
3. The signal processing gain from the maximal-length sequence correlation is not as high as expected. This appears to be due to Doppler shifting of the arrival signal by the change in path length (by surface waves) over the 1.9375 second period. A first order correction for the period for Doppler shift may increase the signal gain.
4. The system has a break in timing synchronization every six hours as the recording tapes are changed. The time code presently used does not have the ± 1 millisecond accuracy for aligning the series. If a reliable method of appending arrival time estimation series together can be found, observations of longer period oscillations (greater than 6 hours) can be made and measurements spanning the tape switch can be made.

5. The present program for estimating arrival times of tomography signals only considers one peak at a time. If the program was modified to choose several it would be more efficient. When this estimation is effectively tied to a certain signal-to-noise ratio threshold, the program could be optimized for real-time operation. The data storage savings for storing only the peaks and not the data would be enormous. Some work has been done in this area but not for continuous transmission of the code.

6. If done by Discrete Fourier Transform (DFT), the code correlation sees all the noise at once. Using the Fast Hadamard Transform (FHT) the correlation sees the noise of one quarter of the signal at a time and interleaves the results of four transforms for the whole correlation. Because of the strict frequency bandlimiting, each interleave of the Hadamard Transform sees almost the same noise, but not quite. Comprehensive comparison of the effect of noise on correlations by the FHT versus DFT should determine any unexpected problems.

7. Another pulse compression scheme, the Gold codes, are related to maximal-length sequences. Gold codes are formed by shifting and adding maximal-length sequences. The codes could be used simultaneously from two different tomographic transmitters because of their low correlation peaks between different (selected) signals.[Ref. 23] Presently they are correlated using DFT's. If an adaptation of the Fast Hadamard Transform algorithm can be applied to their correlation, a large gain in processing efficiency will be realized.

8. The system used in this experiment could perform the code correlations for two channels in real-time. The limiting factor at the present time is in writing the data to disk. With improvements in the speed of writing or a write spooling system, more channels could be matched-filtered in real-time.

9. The Fast Hadamard Transform is only a simplified version of the Fast Fourier Transform with shuffling of the input and output order. The fastest implementation of the FHT may be a hardware solution like those used with FFT's. The reassignment of the serial input to different starting positions should not be difficult, since the positions remain constant. In the same way the output permutation could be done, and the result stored or sent to a computer for further processing.

Finally, the inversion of the travel time fluctuations to map the interior mesoscale fields is still the end goal. All the work on determining the data kernels and system resolution using three-dimensional ray tracing and three-dimensional inverse methods remains. The editing and analysis of data from each transmitter–receiver path will require a large amount of effort to complete, but the edited and processed data set should provide quite a unique opportunity to learn something about the ocean circulation in Monterey Bay.

APPENDIX A

CHRONOLOGIC SUMMARY OF EVENTS IN THE 1988 MONTEREY BAY EXPERIMENT

The following is a summary of the experiment as it happened from the deck log of R/V Point Sur [Ref. 22]. All dates and times are in Pacific Standard Time(PST).

A. 12 DECEMBER 1988

0950 R/V Point Sur underway from Moss Landing. Receiver van is in place on Huckleberry Hill.

1150 Deployed modified AN/SSQ-57 buoy at station B, $36^{o}56.3N$-$122^{o}00.5W$

1241 Deployed MIUW buoy, station B1, $36^{o}36.3N$-$122^{o}00.2W$

1705 Deployed transmitter in 870 meters of water $36^{o}23.7N$-$122^{o}17.84W$

2013 CTD measurement 36°23.2N-122°17.8W

2204 CTD measurement to 1800 meters 36°31.9N-122°17.8W

B. 13 DECEMBER 1988

0013 CTD measurement to 155 meters 36°40.4N-122°04.5W

0105 CTD measurement to 1400 meters 36°40.4N-122°04.4W

0230 Lost contact with buoy at station B

0308 CTD measurement to 73 meters 36°48.6N-122°57.9W

0357 CTD measurement to 800 meters 36°46.5N-122°05.6W

0507 CTD measurement to 800 meters 36°44.7N-122°13.3W

0811 Deployed ARGOS wave buoy #6249 at 36°44.3N-122°13.3W but recovered buoy after no radio signal was received

1033 Deployed modified sonobuoy, station L, 36°52.9N-122°10.8W in 61 fathoms of water

1151 Deployed modified sonobuoy, station J, 36°51.1N-122°04.8W in 53 fathoms of water

1157 CTD measurement to 82 meters 36°51.0N-122°01.5W

1245 Deployed modified sonobuoy, station I, 36°49.1N-122°01.5W in 53 fathoms of water

1339 Deployed modified sonobuoy, station H, 36°51.8N-122°57.2W in 50 fathoms of water

1346 CTD measurement to 73 meters 36°48.5N-121°57.2W

1452 Deployed modified sonobuoy, station G, 36°48.5N-121°57.9W in 53 fathoms of water

1558 Deployed modified sonobuoy, station E, 36°48.5N-121°52.1W in 45 fathoms of water

1605 CTD measurement to 52 meters 36°43.6N-122°00.6W

1713 Deployed ARGOS waves buoy 36°43.6N-122°00.6W

1805 Deployed ARGOS waves buoy 36°43.9N-122°08.6W. Because of the weather forecast for high winds and seas, a decision was made not to deploy the ARGOS thermistor string buoys.

1900 CTD "yo-yo"measurements to 600 meters 36°44.1N-122°13.7W

2200 Stop CTD to reposition - have drifted to 36°43.2N-122°14.7W

2245 CTD "yo-yo"measurements to 600 meters 36°44.5N-122°13.3W

C. 14 DECEMBER 1988

0000 Continue CTD "yo-yo"measurements 36°44.7N-122°14.7W

0033 Halt CTD to move ship (traffic avoidance)

0052 Resume CTD "yo-yo" to 600 meters 36°14.5N-122°13.3W

0338 Stop CTD to reposition - have drifted to 36°45.9N-122°16.7W

0408 CTD "yo-yo" measurements to 600 meters 36°44.6N-122°13.5W

0557 Stop CTD measurements - have drifted to 36°45.2N-122°14.8W

0832 Returned to Moss landing to offload 3 ARGOS buoys and 2 personnel. Remain in port about two hours.

1246 Replace station J modified sonobuoy (replaced with malfunctioning buoy repaired by changing hydrophone, original J buoy recovered)

1435 Deployed MIUW buoy at station L-1 (repaired by splicing power connection in electronics package) 36°55.1N-122°14.0W

1528 Deployed modified sonobuoy at station L-2 (repair unsuccessful and buoy recovered at 1643)

1542 CTD measurement to 80 meters 36°57.6N-122°17.7W

1738 CTD measurement to 90 meters 36°52.8N-122°10.7W

1854 CTD measurement to 1000 meters 36°42.9N-122°13.7W

2046 CTD measurement to 1500 meters 36°32.9N-122°16.7W

2238 CTD measurement to 800 meters 36°23.6N-122°17.9W

D. 15 DECEMBER 1988

0055 CTD "yo-yo" measurement to 600 meters 36°39.0N-122°18.0W

0411 Stop CTD to reposition - have drifted to 36°39.4N-122°23.2W. Winds exceed 40 knots for much of the night.

0445 CTD "yo-yo" measurements to 600 meters 36°38.8N-122°18.2W

0617 CTD to 1200 meters

0644 Stop CTD - have drifted to 36°39.2N-122°22.4W

1138 Recovered ARGOS wave buoy. Begin search for second buoy. Positions are inexact due to three hour time lag in position

report to ship. Swell height limits buoy visibility to about 700 meters.

1623 Discontinue search for ARGOS buoy.

1853 Recovered MIUW buoy, station L-1

2007 Recovered buoy, station L

2114 Recovered buoy, station J

2148 Recovered buoy, station I

2226 Recovered buoy, station h

2257 Recovered buoy, station G

2330 Recovered buoy, station E

E. 16 DECEMBER 1988

0134 Recovered MIUW buoy, station B1, Buoy for station B is not in place

0224 Stop search for station B buoy

0335 CTD measurement to 800 meters $36^{o}30.6$N-$122^{o}09.7$W

0704 Transmitted release signal to acoustic releases on tomography transmitter, no transponder reply heard.

0805 Leave area of tomography transmitter to look for ARGOS buoy.

1030 Recover ARGOS buoy

1253 Transmitted release signal to acoustic releases, which released the anchor.

1331 Transmitter on surface

1421 Transmitter recovered

1830 Moored, Moss Landing

F. DATA DISPOSITION

1. CTD and ADCP data to Woods Hole Oceanographic Institution for Processing.

2. Tomographic acoustic signal recordings to Naval Postgraduate School for processing.

3. NDBC and ARGOS buoy data to National Data Buoy Center for processing.

APPENDIX B

Maximal-length Sequences and the Fast Hadamard Transform

A. INTRODUCTION

Impulsive excitation is an extremely easy and useful mathematical tool for measuring the impulse response of a system or determining travel time through a media. The problem is that an impulse is fairly difficult to achieve physically. As the transmitted pulse approaches an impulse, the required bandwidth and peak power of the transmitter increase. Impulsive sound signals can be generated by explosive or implosive sources but these have uneven frequency distribution energy, and repeatability. Another solution is to use pseudorandom noise. The period, frequency distribution, and energy are deterministic and can be tailored to meet system requirements. The signal can be repeated identically for additional signal processing gain. Importantly, when sampled and digitized the signal becomes an impulse of much shorter duration and higher peak power than the original signal. The method for generating the sequence as well as a fast method for processing the received signal will be described here.

The pseudorandom noise signal is a binary maximal-length shift register sequence. The sequence's most important characteristic its autocorrelation, which is constant except at a shift of zero, making the sequence the equivalent of white noise. The energy at zero is much higher than for each individual digit, making it easier to estimate the arrival time of the signal. Other

properties of maximal-length sequences (m-sequences) are detailed by Ziemer and Peterson [Ref. 24], including a list of polynomials which produce m-sequences. Not all shift register sequences are of maximal-length, only those which do not repeat until after 2^n-1 delays, where n is the number of delays in the shift register.

As an example the table entry for a maximal-length sequence of degree three will be developed into a code and a fast method for its autocorrelation will be examined. Various sources were used.[Refs. 25,26,27]

B. GENERATING THE M-SEQUENCE

Table 8-5 of Ziemer and Peterson [Ref. 24] lists only one polynomial for generating an m-sequence of degree three. The length of the sequence will be seven digits. The listing in the table is an octal representation of the binary coefficients of the generating polynomial. Translating to binary this becomes

$$[\,1\,3\,]_8 \quad ==> \quad [\,1\,0\,1\,1\,]_2\ . \tag{B.1}$$

The corresponding polynomial is

$$g(D) = D^3 + D + 1\ , \tag{B.2}$$

where D is a delay of one unit (D^3 is three delays). The shift register register realization follows directly as shown in Figure 21.

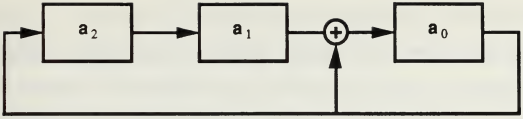


Figure 21: Shift register realization of equation (B.2).

Loading the initial state is arbitrary since the register will cycle through all possible combinations before repeating. For an initial state $a_2=1$, $a_1=0$, $a_0=0$, this is one period of the sequence as shown in Figure 22.

Cycle	a_2	a_1	a_0
1	1	0	0
2	0	1	0
3	0	0	1
4	1	0	1
5	1	1	1
6	1	1	0
7	0	1	1
8	1	0	0

Figure 22: Shift register contents when generating M-sequence. Note that the eighth cycle is only included to show that the register begins to repeat.

The m-sequence is a single column of the register states. The characteristics of the autocorrelation are unaffected by whether the m-sequence is read from top to bottom or the reverse, but the method for formulating the Hadamard demodulation does change. The top to bottom sequence will be designated the "forward" code,

forward code $S = 1001110$,

reverse code $S = 0111001$. (B.3)

In use, the m-sequence digits are transformed by replacing 1 with -1 and 0 with 1. When dealing with the structure and mathematics it is easier to use 0 and 1 because many people are familiar with binary mathematics and can more easily adapt to modulo-two mathematics.

The received signal has an unknown time delay and so must be correlated with all possible shifts of the code. Let the seven shifted sequences form the matrix M:

$$M = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix} . \quad \text{(B.4)}$$

When this matrix and the code are transformed to + and -1's , multiplying the signal by the matrix will result in the correlation,

$$R_{sm} = MS \ . \quad \text{(B.5)}$$

This is the entire goal of the initial signal processing, all that remains is to develop a fast, efficient algorithm to accomplish this multiplication.

C. THE HADAMARD MATRIX

To describe the fast algorithm, it is necessary to introduce the Hadamard matrix. The Sylvester-type Hadamard Matrix has a recursive form for higher orders given by

$$\mathbf{H}_1 = [1], \qquad \mathbf{H}_{2i} = \begin{bmatrix} \mathbf{H}_i & \mathbf{H}_i \\ \mathbf{H}_i & -\mathbf{H}_i \end{bmatrix}. \tag{B.6}$$

The third degree matrix $\mathbf{H}$ is

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}, \tag{B.7}$$

or, represented by ones and zeros,

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \qquad \text{(B.8)}$$

One way to form the matrix is by multiplying matrices formed of the binary 'counting' matrix from 0 to 7,

$$H = A A^T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}. \qquad \text{(B.9)}$$

The matrix M can be factored in the same fashion, but not as simply. Form the first matrix B from the successive contents of the shift register, but bit reversed (from right to left)and in reverse order (from bottom to top). The original order is then preserved by shifting the rows of the matrix to bring the 3x3 identity matrix to the top,

$$B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} . \qquad \text{(B.10)}$$

Form the second matrix **C** from three shifted versions of the m-sequence

$$C = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} . \qquad \text{(B.11)}$$

It is easy to verify that

$$BC = M. \qquad \text{(B.12)}$$

Note that M,B, and C matrices must be expanded by a leading row and/or column of zeros to be of the proper size. The new matrices will be denoted with a prime. If mapping matrices can be found such that $QA = B'$ and $A^tP = C'$ then the same matrices will map the Hadamard matrix to the m-sequence matrix

$$M' = B'C' = QAA^tP = QHP . \qquad \text{(B.13)}$$

Recall that the correlation for the signal with the output code is given by multiplication,equation (B.5), which now becomes

$$R'_{sm} = M'S' . \qquad \text{(B.14)}$$

(S' because the leading zeros must be added.) Combining equations (B.13) and (B.14) results in

$$R'_{sm} = QHPS' \ . \qquad (B.15)$$

This gives the signal correlation that is required. The initial entry is removed to change R'_{sm} to R_{sm} .

D. INPUT AND OUTPUT VECTOR ORDER PERMUTATION

The matrices P and Q must be found such that QA = B' and $A^tP = C'$. A natural index for each row or column is its equivalent octal value since the values range from 0 to 7 and do not repeat as shown in Figure 23.

$$A = \begin{bmatrix} 0&0&0 \\ 0&0&1 \\ 0&1&0 \\ 0&1&1 \\ 1&0&0 \\ 1&0&1 \\ 1&1&0 \\ 1&1&1 \end{bmatrix} \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{matrix} \qquad B' = \begin{bmatrix} 0&0&0 \\ 1&0&0 \\ 0&1&0 \\ 0&0&1 \\ 1&1&0 \\ 0&1&1 \\ 1&1&1 \\ 1&0&1 \end{bmatrix} \begin{matrix} 0 \\ 4 \\ 2 \\ 1 \\ 6 \\ 3 \\ 7 \\ 5 \end{matrix}$$

$$A^T = \begin{bmatrix} 0&0&0&0&1&1&1&1 \\ 0&0&1&1&0&0&1&1 \\ 0&1&0&1&0&1&0&1 \end{bmatrix} \qquad C' = \begin{bmatrix} 0&1&0&0&1&1&1&0 \\ 0&0&1&0&0&1&1&1 \\ 0&1&0&1&0&0&1&1 \end{bmatrix}$$

$$0\ 1\ 2\ 3\ 4\ 5\ 6\ 7 \qquad 0\ 5\ 2\ 1\ 4\ 6\ 7\ 3$$

Figure 23: Indices formed from matrix octal equivalents.

The permutation matrices will have ones in the following positions:

Q row	0	1	2	3	4	5	6	7
Q column	0	4	2	1	6	3	7	5
P row	0	5	2	1	4	6	7	3
P column	0	1	2	3	4	5	6	7 .

(B.16)

These indices are important. With the indices, the matrices do not have to be constructed. The 'multiplication' by the permutation matrices is accomplished by shuffling the order of the signal vector, rather than direct multiplication, as shown in Figure 24. Note that no multiplications are required, only the reordering. For a given code the permutations can be evaluated once and the result stored as an index array to be applied to each vector.

E. THE FAST HADAMARD TRANSFORM

There exists an efficient method of performing the multiplication by the Hadamard matrix. If a vector is multiplied by the Hadamard matrix (the normal Hadamard matrix of {+1,-1}). The result is a vector of sums of all the components of the vector with various + and - weighting.

P column	P row	S' becomes	PS'
0	0	S_0	G_0
1	5	S_1	G_5
2	2	S_2	G_2
3	1	S_3	G_1
4	4	S_4	G_4
5	6	S_5	G_6
6	7	S_6	G_7
7	3	S_7	G_3

Q row	Q column	F=HPS' becomes	R=QHPS'
0	0	F_0	R_0
1	4	F_4	R_1
2	2	F_2	R_2
3	1	F_1	R_3
4	6	F_6	R_4
5	3	F_3	R_5
6	7	F_7	R_6
7	5	F_5	R_7

Figure 24: Re-ordering of input and output vectors according to the permutation matrices P and Q. For the input vector, let G=PS' and for the output vector, let F=HPS'.

Define a vector **V** such that

$$\mathbf{V} = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} . \qquad \text{(B.17)}$$

After multiplying this by the Hadamard matrix the vector becomes

$$HV = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix}$$

$$= \begin{bmatrix} a+b+c+d+e+f+g+h \\ a-b+c-d+e-f+g-h \\ a+b-c-d+e+f-g-h \\ a-b-c+d+e-f-g+h \\ a+b+c+d-e-f-g-h \\ a-b+c-d-e+f-g+h \\ a+b-c-d-e-f+g+h \\ a-b-c+d-e+f+g-h \end{bmatrix} . \qquad \text{(B.18)}$$

When calculating correlations, let a=0 so that no new information is added. The zeroth position result is the sum of all the elements of the code and is therefore equal to the DC pedestal. This pedestal can be removed by subtracting this sum of all elements, or not, depending on the application.

Compare this result to the result using a flow diagram identical to the procedure used with the Fast Fourier Transform, except that all the 'twiddle' factors are equal to one, Figure 25.

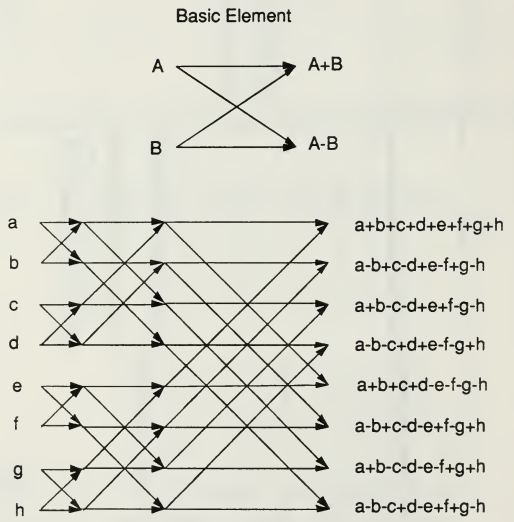


Figure 25: Basic Fast Hadamard Transform element for cascading additions and the full diagram for an eight point FHT.

The result of the Fast Hadamard Transform is the same as for multiplication. The algorithm used for the Fast Fourier Transform is

trivialized in this case - there is no bit reversal or multiplication by a phase factor. Because the method requires only additions, the exact computational speed increase is difficult to calculate. (The speed improvement for FFT over DFT is usually calculated by comparing the number of multiplications required) The 'multiplication' by P and Q has been replaced by reordering, so that there is no multiplication required. The speed of execution now depends on other statements in the program as well as the correlation because loop increments and tests for completion may take as long as the additions.

F. USING THE REVERSE CODE

The permutation matrices for the reverse code are found in a slightly different way. The matrix B is found from the contents of the shift register directly, not bit reversed and in reverse order as for the forward code. The matrix C is formed by shifting the code to the left (vice right). The permutation indices are determined and used in the same fashion as before.

G. CORRELATION PROCEDURE

The procedure for performing the correlation can now be summarized in five straightforward steps:

1. Augment the signal vector S by adding a zero in the zeroth position.
2. Permute the vector according to P.
3. Perform the Fast Hadamard Transform.
4. Permute the resulting vector according to Q.
5. Remove the zeroth entry.

H. EXAMPLE

Consider the first and third rows of the m-sequence matrix as input signals:

first							third						
1	0	0	1	1	1	0	1	0	1	0	0	1	1

Transform to {-1,+1} The result is the signal vector **S** as would be received.

-1	1	1	-1	-1	-1	1	-1	1	-1	1	1	-1	-1

Add beginning 0

0	-1	1	1	-1	-1	-1	1	0	-1	1	-1	1	1	-1	-1

Permute according to **P**

0	1	1	1	-1	-1	-1	-1	0	-1	1	-1	1	-1	1	-1

Perform Fast Hadamard Transform (can be done in this case by comparing to rows in the Hadamard matrix for a match)

-1	-1	-1	-1	7	-1	-1	-1	-1	7	-1	-1	-1	-1	-1	-1

Permute according to **Q**

-1	7	-1	-1	-1	-1	-1	-1	-1	-1	-1	7	-1	-1	-1	-1

Remove the zeroth element

-7	-1	-1	-1	-1	-1	-1	-1	-1	7	-1	-1	-1	-1

As expected, the correlation produces a peak in the first and third positions, respectively.

I. SUMMARY

When performing the correlation of a signal and the m-sequence using the Fast Hadamard Transform and a quadrature demodulation system the real and imaginary components of the signal are correlated separately and later combined for magnitude and phase. Note that the FHT only works on one sample per digit of the m-sequence in the signal. For improved accuracy in estimating the arrival time of the impulse, it is valuable to sample at a higher frequency. This may also allow digital filtering. The sampling frequency should be an integer multiple of the code clock rate (also known as the "chip" rate). The data samples should then be decimated into records at the code clock frequency so that they are again one sample per digit. After the FHT correlation the data interleaves are recombined to their original positions. For example, if a code clocked at 16 Hz is sampled at 64 Hz, then 4 separate correlations will have to be performed on each of the in-phase and quadrature channels. The FHT correlation is still much faster than using DFT or FFT methods, or matrix multiplies.

The result of the FHT correlation in the case of data sampled at higher than the code clock rate is not the same as for conventional correlation. In an ideal case, each of the interleaves will produce an output peak of equal magnitude, resulting in a 'flat-topped' correlation peak, vice a 'pointy' correlation peak. The estimation of travel time must look for this shape, rather than the 'point'.

APPENDIX C

SIGNAL PROCESSING PROGRAMS

The following programs are listed in this appendix:

A. AMORE - Data conversion and code correlation

B. AINPUT - Data conversion

C. AHAD - Code correlation

D. ACRID - Coherent averaging, arrival correlation, and arrival time estimation.

E. AGONY - Coherent averaging, arrival correlation, and interactive arrival time estimation.

F. AGRAF4 - Generate graphics files for use with MATLAB.

G. AGRAF5 - Generate graphics files for use with SURFER.

A. PROGRAM AMORE

This program makes use of a MetraByte DASH16F data aquisition and control interface board in conjunction with the data synchronous quadrature demodulator to digitize and perform the maximal-length sequence correlation on four input channels (in-phase and quadrature for two stations) with an external interrupt for triggering the data conversion. The program makes extensive use of the routines DASH16, SEGADR, OFFADR, SEGPTR, OFFPTR, all provided with the DASH16f by the manufacturer in the library file DAS16F.LIB.

The data is converted and stored in either of two memory buffers using direct memory access (DMA) on the Zenith Z-200 AT. This leaves the program free to conduct the code correlation on the data in the other memory buffer using the Fast Hadamard Transform as described in Appendix B . After correlation the data is written to disk. The same data is then coherently averaged for 16 code periods and this data is written to a separate disk file. This program was typically used for converting 6 hours of two channels of data recorded on videotape to two Bernoulli Box 20 Mbyte cartridges. This converted to about 17 Mbytes of data for each channel.

```
*       TOMOGRAPHIC SIGNAL DIGITIZATION PROGRAM
*
*  DATA INPUT FOR FOUR CHANNELS WITH 64 HZ EXTERNAL TRIGGER
*       DATA IS INTIALLY TRANSFERRED TO MEMORY BY DMA THEN
*       BROUGHT TO THE FORTRAN PROGRAM FOR CODE CORRELATION AND
*       WRITING TO THE BERNOULLI BOX.
*       Microsoft FORTRAN77
*         On compilation, program must be linked to DAS16F.LIB which
*         is provided with the DAS16F A/D board.
*       BOB DEES  3 FEB 89

      INTEGER*2 PARAM(10), DATA(7936), CHAN1(3968), CHAN2(3968)

      INTEGER*2 BASE,INTLEV,DMALEV,TCHAN(248),TMAG(124)

      INTEGER*2 MAG1(1984), MAG2(1984), PHASE1(1984), PHASE2(1984)

      INTEGER*2 MODE, RCODE, HOUR, MINUTE, CNVTIM, TIME, SEC
```

```
      INTEGER*2 DASH16, SEGADR, OFFADR, SEGPTR, OFFPTR
      INTEGER*2 TRIG, RCYC, NOC, NOS, DISP, INCR,,TPHAS(124)
      INTEGER*2 I, J, K, N, AMAG(124), APHASE(124)
      INTEGER*2 SCH, FCH
      INTEGER*4 BUFFER, ALLOC
      CHARACTER*20 NAME1$,NAME2$,NAME3$,NAME4$
      CHARACTER*1 DOT(62,20),ANS$
      CHARACTER*8 CHAN1$, CHAN2$, DATE$, STATN1$, STATN2$
115   FORMAT (' Invalid Input.  Please, try again !')
120   FORMAT (' Mode ', I2, ' ,  Error = ', I4)
124   FORMAT (' Cannot Allocate Buffer')
127   FORMAT (A1)
129   FORMAT (A20)
131   FORMAT ('POSIT=',A10,' STATION=',A10,'DATE=',A10)
132   FORMAT (A8)
133   FORMAT (I6)
134   FORMAT (2I3)
135   FORMAT ('TIME IS: HOUR= ',I3,'  MINUTE=',I3)
136   FORMAT (1X,'FINISHED WRITING TOP TO DISK, TIME=',
     Z    I3,' HR',I3,' MIN',I3,' SEC')
137   FORMAT (1X,'FINISHED WRITING BOTTOM TO DISK, TIME=',
     Z    I3,' HR',I3,' MIN',I3,' SEC')
138   FORMAT (A45)
171   FORMAT(' CONVERSION # =',I6)
180   FORMAT(2I5)
*  INPUT INFORMATION ABOUT SIGNALS
      WRITE(*,*)' Welcome to the tomographic data input program!'
```

```
      write(*,*)char(13),'Want more info? (Y/N)'
      read(*,127)ans$
      if(ans$.eq.'N'.or.ans$.eq.'n')goto  200
      write(*,*)char(13),'This prgram uses the DAS16F A/D board'
      write(*,*)'for sampling data with an external interrupt.'
      write(*,*)'The in-phase and quadrature signals for one signal'
      write(*,*)'should be connected to chan. 0 and 1. The second'
      write(*,*)'should be connected to chan. 2 and 3. Note that the'
      write(*,*)'program asks for the time length of conversion. This'
      write(*,*)'time is figured from the time synchronization signal'
      write(*,*)'supplying the interrupts and if it stops before the'
      write(*,*)'end of the conversion, the program will continue on'
      write(*,*)'with the count provided by the free running PLL. If'
      write(*,*)'this signal is lost or an end of conversion is needed'
      write(*,*)'use soft restart(ctrl,c).DANGER*** HARD RESTART '
      write(*,*)'WILL DUMP THE DATA! DO NOT PRESS(CTRL,ALT,DEL).'
      write(*,*)'Before writing to disk the program performs the M code'
      write(*,*)'correlation and converts the result to magnitude and'
      write(*,*)'phase. Additionally a file of data coherently averaged'
      write(*,*)'by 16 periods is written to disk.'

200   continue
      WRITE(*,*)CHAR(13),'WHAT POSITION FOR CHANNEL 1 ?'
      WRITE(*,*)'EXAMPLE: L'

      READ(*,132)STATN1$

      WRITE(*,*)'CHANNEL 1 RADIO TRANSMITTER CHANNEL ?'
      WRITE(*,*)'EXAMPLE: 29'

      READ(*,132)CHAN1$

      WRITE(*,*)'WHAT POSITION FOR CHANNEL 2 ?'

      READ(*,132)STATN2$

      WRITE(*,*)'CHANNEL 2 RADIO TRANSMITTER CHANNEL ?'

      READ(*,132)CHAN2$

      WRITE(*,*)'WHAT IS THE DATE ?'
      WRITE(*,*)'EXAMPLE: 12DEC88'

      READ(*,132)DATE$

      WRITE(*,*)'BEGINNING TIME (HOUR,MINUTE)'
      WRITE(*,*)'MUST BE INTEGERS, EXAMPLE: 18,59'

      READ(*,134)HOUR,MINUTE

      WRITE(*,*)'FILENAME FOR CHANNEL 1 ?'
```

```
   WRITE(*,*)'FORMAT FOR FILENAME IS C:L1218.OUT. L FOR POSIT,'
   WRITE(*,*)'12 FOR 12DEC88, 18 FOR BEGINNING TIME 18TH HOUR, '
   WRITE(*,*)'OUT FOR CORRELATED DATA, D IS THE BERNOULLI
Z  DRIVE.'
   WRITE(*,*)'  EXAMPLE: D:L1218.OUT'

   READ(*,129)NAME1$

   WRITE(*,*)'FILENAME FOR COHERENTLY AVERAGED OUTPUT(FOR
Z DISPLAY'
   WRITE(*,*)'USING AGRAF2 AND MATLAB) EXAMPLE: D:L1218.MAG'

   READ(*,129)NAME3$

   WRITE(*,*)'FILENAME FOR CHANNEL 2 ?'

   READ(*,129)NAME2$

   WRITE(*,*)'FILENAME FOR COHERENTLY AVERAGED OUTPUT(FOR
 Z  DISPLAY'
   WRITE(*,*)'USING AGRAF2 AND MATLAB) FOR CHANNEL 2'

   READ(*,129)NAME4$

   WRITE(*,*)'HOW MANY MINUTES OF CONVERSION TIME ?'
   WRITE(*,*)'TEN MINUTES = .47 MBYTES/CHANNEL'

   READ(*,133)CNVTIM
   TIME=0

   OPEN(33,FILE=NAME1$)
   OPEN(34,FILE=NAME2$)
   OPEN(35,FILE=NAME3$)
   OPEN(36,FILE=NAME4$)
         rewind 33
         rewind 34
         rewind 35
         rewind 36

         WRITE(33,131)STATN1$,CHAN1$,DATE$
         WRITE(34,131)STATN2$,CHAN2$,DATE$
         WRITE(35,131)STATN1$,CHAN1$,DATE$
         WRITE(36,131)STATN2$,CHAN2$,DATE$
         WRITE(33,135)HOUR,MINUTE
         WRITE(34,135)HOUR,MINUTE
         WRITE(35,135)HOUR,MINUTE
         WRITE(36,135)HOUR,MINUTE
         WRITE(35,138)'COHERENTLY AVERAGED Yx16 BLOCK
Z CORRELATION N'
         WRITE(36,138)'COHERENTLY AVERAGED Yx16 BLOCK
Z CORRELATION N'
```

```
*   Use mode 0 to initialize DASH-16 Board

        MODE = 0
        PARAM(1) = #300
        PARAM(2) = 2
        PARAM(3) = 1
        RCODE = DASH16(MODE, PARAM)
        IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE

*   FIX MEMORY BUFFER

        BUFFER = ALLOC(15872)
*   15872=64HZ*62SEC*4CHANNELS
        IF (BUFFER .EQ. 0) WRITE(*, 124)

*   EXTERNAL TRIGGER

        TRIG=0

*   REUSE SAME MEMORY AREA

        RCYC=0

*   SET A/D CHANNEL SCAN LIMITS(0-3)

        SCH=0
        FCH=3
        N = FCH - SCH + 1

*       Set Number of CONVERSIONS(31 SEC)

        NOC = 7936

600     CONTINUE

*       Use mode 1 to load Queue

        MODE = 1
        PARAM(1) = SCH
        PARAM(2) = FCH
        RCODE = DASH16(MODE, PARAM)

        IF (RCODE .NE. 0) WRITE(*, 120)MODE, RCODE

*       Perform 31 SEC OF CONVERSIONS FOR INPUTS 0-3
*   DMA TO BOTTOM OF BUFFER

610     CONTINUE
```

```
      MODE = 20
      PARAM(1) = NOC
      PARAM(2) = SEGPTR(BUFFER)
      PARAM(3) = TRIG
      PARAM(4) = RCYC
      RCODE = DASH16(MODE, PARAM)

      IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE
       IF(TIME.EQ.0.AND.SEC.EQ.0)GOTO 650

*  DATA TRANSFER, MODE 9

      WRITE(*,*)'DONE WITH DMA TO TOP'
      MODE=9
      PARAM(1)=7936
      PARAM(2)=SEGPTR(BUFFER)+7936
      PARAM(3)=0
      PARAM(4)=OFFADR(DATA)
      PARAM(5)=0
      PARAM(6)=0
      RCODE=DASH16(MODE,PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
      DO 620 I=1,3967,2
      CHAN1(I)=DATA(2*I-1)
            CHAN1(I+1)=DATA(2*I)
            CHAN2(I)=DATA(2*I+1)
      CHAN2(I+1)=DATA(2*I+2)
620   CONTINUE

      DO 645 I= 0,15
            DO 625 J=1,248
                  TCHAN(J)=CHAN1(I*248+J)
625         CONTINUE
            CALL CORREL(TCHAN,TMAG,TPHAS)
            DO 627 J=1,124
                  MAG1(I*124+J)=TMAG(J)
                  PHASE1(I*124+J)=TPHAS(J)
627         CONTINUE
            DO 628 J=1,248
                  TCHAN(J)=CHAN2(I*248+J)
628         CONTINUE
            CALL CORREL(TCHAN,TMAG,TPHAS)
            DO 629 J=1,124
                  MAG2(I*124+J)=TMAG(J)
                  PHASE2(I*124+J)=TPHAS(J)
629         CONTINUE
645   CONTINUE

      CALL AVG(MAG1,PHASE1,AMAG,APHASE)
      WRITE(35,180)(AMAG(I),APHASE(I),I=1,124)
      WRITE(33,180)(MAG1(I),PHASE1(I),I=1,1984)
```

```
      CALL AVG(MAG2,PHASE2,AMAG,APHASE)
      WRITE(36,180)(AMAG(I),APHASE(I),I=1,124)
      WRITE(34,180)(MAG2(I),PHASE2(I),I=1,1984)

            SEC=SEC+31
            IF(SEC.GT.60)THEN
                  TIME=TIME+1
                  MINUTE=MINUTE+1
                  SEC=SEC-60
            ENDIF
            IF(MINUTE.GE.60)THEN
                  MINUTE=MINUTE-60
                  HOUR=HOUR+1
                  IF(HOUR.EQ.24)HOUR=0
            ENDIF
            WRITE(*,136)HOUR,MINUTE,SEC
      IF(TIME.GE.CNVTIM)GOTO 999
            WRITE(*,*)' '
            WRITE(*,*)' '
            WRITE(*,*)'######  channel 2  ######### '
                                    DO 647 DX=1,62
                                          DO 646 DY=1,20
                                          DOT(DX,DY)=' '
646                                       CONTINUE
647                                 CONTINUE
                                    DO 648 II=2,124,2
                                          DX=II/2
                                          DY=MAG2(II)/25
                                          IF(DY.GT.20)DY=20
                                          IF(DY.LT.1)DY=1
                                          DOT(DX,DY)='*'
648                                 CONTINUE
                                    DO 649 DY=20,1,-1
                                    WRITE(*,*)(DOT(DX,DY),DX=1,62)
649                                 CONTINUE



*     use mode 8 to obtain status of CONVERSION
*  RETURNS OPERATION, STATUS, CURRENT WORD COUNT
*   AS PARAMETERS

650   CONTINUE

      MODE = 8
      PARAM(3) = 0
      RCODE = DASH16(MODE, PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
      IF(PARAM(3).EQ.7936)then
      write(*,*)'FAILURE IN BOTTOM WRITE-TOO SLOW',CHAR(7),CHAR(7)
```

```
            GO TO 999
        ENDIF

700     CONTINUE

        RCODE = DASH16(MODE, PARAM)
        IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
        IF(PARAM(3).LE.7936)GOTO 700

*  HALT DMA, MODE 7

        MODE=7
        RCODE=DASH16(MODE,PARAM)
        IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE

*  RESTART CONVERSIONS , DMA TO TOP OF BUFFER

        MODE = 20
        PARAM(1) = NOC
        PARAM(2) = SEGPTR(BUFFER)+7936
        PARAM(3) = TRIG
        PARAM(4) = RCYC
        RCODE = DASH16(  MODE, PARAM)

        IF(RCODE.NE.0)WRITE(*,120)MODE,RCODE
        Write(*,*)'DONE WITH DMA TO BOTTOM'

*       DATA TRANSFER, MODE 9
*  PARAMETERS 1-# OF WORDS TO TRANSFER, 2-SOURCE SEGMENT
*  3-STARTING CONVERSION NUMBER, 4-DATA ARRAY, 5-CHANNEL ARRAY

        MODE=9
        PARAM(1)=7936
        PARAM(2)=SEGPTR(BUFFER)
        PARAM(3)=0
        PARAM(4)=OFFADR(DATA)
        PARAM(5)=0
        PARAM(6)=0
        RCODE=DASH16(MODE,PARAM)
        IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
        DO 800 I=1,3967,2
        CHAN1(I)=DATA(2*I-1)
        CHAN1(I+1)=DATA(2*I)
        CHAN2(I)=DATA(2*I+1)
        CHAN2(I+1)=DATA(2*I+2)
800     CONTINUE

        DO 845 I= 0,15
                DO 825 J=1,248
                        TCHAN(J)=CHAN1(I*248+J)
825             CONTINUE
```

```
            CALL CORREL(TCHAN,TMAG,TPHAS)
            DO 827 J=1,124
                  MAG1(I*124+J)=TMAG(J)
                  PHASE1(I*124+J)=TPHAS(J)
827         CONTINUE
            DO 828 J=1,248
                  TCHAN(J)=CHAN2(I*248+J)
828         CONTINUE
            CALL CORREL(TCHAN,TMAG,TPHAS)
            DO 829 J=1,124
                  MAG2(I*124+J)=TMAG(J)
                  PHASE2(I*124+J)=TPHAS(J)
829         CONTINUE
845   CONTINUE

      CALL AVG(MAG1,PHASE1,AMAG,APHASE)
      WRITE(35,180)(AMAG(I),APHASE(I),I=1,124)
      WRITE(33,180)(MAG1(I),PHASE1(I),I=1,1984)

      CALL AVG(MAG2,PHASE2,AMAG,APHASE)
      WRITE(36,180)(AMAG(I),APHASE(I),I=1,124)
      WRITE(34,180)(MAG2(I),PHASE2(I),I=1,1984)

            SEC=SEC+31
            IF(SEC.GT.60)THEN
                  TIME=TIME+1
                  MINUTE=MINUTE+1
                  SEC=SEC-60
            ENDIF
            IF(MINUTE.GE.60)THEN
                  MINUTE=MINUTE-60
                  HOUR=HOUR+1
                  IF(HOUR.EQ.24)HOUR=0
            ENDIF
            WRITE(*,*)' '
            WRITE(*,*)' '
            WRITE(*,*)'######  channel 1  ######### '
                              DO 92 DX=1,62
                                    DO 91 DY=1,20
                                    DOT(DX,DY)=' '
91                                  CONTINUE
92                                CONTINUE
                                  DO 94 II=2,124,2
                                    DX=II/2
                                    DY=MAG1(II)/25
                                    IF(DY.GT.20)DY=20
                                    IF(DY.LT.1)DY=1
                                    DOT(DX,DY)='*'
94                                CONTINUE
                                  DO 95 DY=20,1,-1
                                  WRITE(*,*)(DOT(DX,DY),DX=1,62)
```

```
95                              CONTINUE

      WRITE(*,*)'ELAPSED TIME ',TIME,'  MINUTES'
      WRITE(*,137)HOUR,MINUTE,SEC
            IF(TIME.GE.CNVTIM)GO TO 999

*  USE MODE 8 TO GET STATUS OF CONVERSION

      MODE = 8
      PARAM(3) = 0
      RCODE = DASH16(MODE, PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
      IF(PARAM(3).EQ.7936)then
            write(*,*)'FAILURE IN BOTTOM WRITE-TOO SLOW',CHAR(7),
     Z CHAR(7)
            GO TO 999
      ENDIF
900   CONTINUE
      RCODE = DASH16(MODE, PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
      IF(PARAM(3).LE.7936)GOTO 900

*  HALT DMA, MODE 7

      MODE=7
      RCODE=DASH16(MODE,PARAM)
      IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE
      GOTO610
999   CONTINUE
      WRITE(*,*)'  DONE!!!',CHAR(7),CHAR(7),CHAR(7)
      STOP
      END

*************************************************************

      SUBROUTINE CORREL(CHAN,MMAG,PPHASE)
*THIS ROUTINE CALLS DATA IN ARRAY CHAN, PERMUTES THE ORDER,
*CONDUCTS A FAST HADAMARD TRANSFORM, PERMUTES THE DATA AGAIN,
*THEN RETURNS THE DATA IN AS MMAG AND PPHASE. CHAN CONTAINS
*DATA AS IN-PHASE AND QUADRATURE BASEBAND SIGNAL SAMPLES -
*COS(THETA) AND SIN(THETA). THE OUTPUT CONSISTS OF THE MAGNITUDE
*AND PHASE OF THE CORRELATED SIGNAL. BOB DEES 7MAR89.


      INTEGER*2 CHAN(248),PPHASE(124),MMAG(124),DTA(31,4,2)
      INTEGER I,J,K,L,N,M,ISPACE,IWIDTH,ITOP,IBOT,TEMP
      INTEGER DX,DY,II,KK
      INTEGER*4 DCLVL,PDSTAL
      INTEGER*4 PRDATA(0:31,4,2),OUTDTA(31,4,2)
      INTEGER*4 MAG(31,4),PHASE(31,4)
```

```
      CHARACTER*1 DOT(62,20),ANS$

****************** TRANSFER DATA ***************************
      DO 120 I=1,31
            DO 110 K=1,4
                  DO 100 J=1,2
                        IJK=(I-1)*8+(K-1)*2+J
                        DTA(I,K,J)=CHAN(IJK)
100               CONTINUE
110         CONTINUE
120   CONTINUE

****************** PERMUTE *************************
      DO 190 K=1,4
      DO 180 J=1,2
            PRDATA(0,K,J)=0
            PRDATA(10,K,J)=DTA(1,K,J)
            PRDATA(5,K,J)=DTA(2,K,J)
            PRDATA(2,K,J)=DTA(3,K,J)
            PRDATA(1,K,J)=DTA(4,K,J)
            PRDATA(16,K,J)=DTA(5,K,J)
            PRDATA(8,K,J)=DTA(6,K,J)
            PRDATA(4,K,J)=DTA(7,K,J)
            PRDATA(18,K,J)=DTA(8,K,J)
            PRDATA(9,K,J)=DTA(9,K,J)
            PRDATA(20,K,J)=DTA(10,K,J)
            PRDATA(26,K,J)=DTA(11,K,J)
            PRDATA(13,K,J)=DTA(12,K,J)
            PRDATA(6,K,J)=DTA(13,K,J)
            PRDATA(19,K,J)=DTA(14,K,J)
            PRDATA(25,K,J)=DTA(15,K,J)
            PRDATA(28,K,J)=DTA(16,K,J)
            PRDATA(30,K,J)=DTA(17,K,J)
            PRDATA(31,K,J)=DTA(18,K,J)
            PRDATA(15,K,J)=DTA(19,K,J)
            PRDATA(7,K,J)=DTA(20,K,J)
            PRDATA(3,K,J)=DTA(21,K,J)
            PRDATA(17,K,J)=DTA(22,K,J)
            PRDATA(24,K,J)=DTA(23,K,J)
            PRDATA(12,K,J)=DTA(24,K,J)
            PRDATA(22,K,J)=DTA(25,K,J)
            PRDATA(27,K,J)=DTA(26,K,J)
            PRDATA(29,K,J)=DTA(27,K,J)
            PRDATA(14,K,J)=DTA(28,K,J)
            PRDATA(23,K,J)=DTA(29,K,J)
            PRDATA(11,K,J)=DTA(30,K,J)
            PRDATA(21,K,J)=DTA(31,K,J)
180   CONTINUE
190   CONTINUE
****************** FAST HADAMARD*****************
```

```
      DO 300 K=1,4
      DO 290 J=1,2
            DO 270 L=1,5
                  ISPACE=2**L
                  IWIDTH=2**(L-1)
                  DO 250 N=0,(IWIDTH-1)
                        DO 230 ITOP=N,(32-2),ISPACE
                              IBOT=ITOP+IWIDTH
                              TEMP=PRDATA(IBOT,K,J)
                              PRDATA(IBOT,K,J)=PRDATA(ITOP,K,J)-TEMP
                              PRDATA(ITOP,K,J)=PRDATA(ITOP,K,J)+TEMP
230                     CONTINUE
250               CONTINUE
270         CONTINUE
290   CONTINUE
300   CONTINUE
************ PERMUTE AND REMOVE BIAS ***************

      DO 340 K=1,4
      DO 330 J=1,2
            DCLVL=(ABS(DTA(1,K,J))-PRDATA(0,K,J))/30
            PDSTAL=PRDATA(0,K,J)+DCLVL*31
* NOTE:DCLVL ISN'T THE SAME AS THE PEDESTAL
            OUTDTA(1,K,J)=PRDATA(1,K,J)-DCLVL-PDSTAL
            OUTDTA(2,K,J)=PRDATA(18,K,J)-DCLVL-PDSTAL
            OUTDTA(3,K,J)=PRDATA(9,K,J)-DCLVL-PDSTAL
            OUTDTA(4,K,J)=PRDATA(22,K,J)-DCLVL-PDSTAL
            OUTDTA(5,K,J)=PRDATA(11,K,J)-DCLVL-PDSTAL
            OUTDTA(6,K,J)=PRDATA(23,K,J)-DCLVL-PDSTAL
            OUTDTA(7,K,J)=PRDATA(25,K,J)-DCLVL-PDSTAL
            OUTDTA(8,K,J)=PRDATA(30,K,J)-DCLVL-PDSTAL
            OUTDTA(9,K,J)=PRDATA(15,K,J)-DCLVL-PDSTAL
            OUTDTA(10,K,J)=PRDATA(21,K,J)-DCLVL-PDSTAL
            OUTDTA(11,K,J)=PRDATA(24,K,J)-DCLVL-PDSTAL
            OUTDTA(12,K,J)=PRDATA(12,K,J)-DCLVL-PDSTAL
            OUTDTA(13,K,J)=PRDATA(6,K,J)-DCLVL-PDSTAL
            OUTDTA(14,K,J)=PRDATA(3,K,J)-DCLVL-PDSTAL
            OUTDTA(15,K,J)=PRDATA(19,K,J)-DCLVL-PDSTAL
            OUTDTA(16,K,J)=PRDATA(27,K,J)-DCLVL-PDSTAL
            OUTDTA(17,K,J)=PRDATA(31,K,J)-DCLVL-PDSTAL
            OUTDTA(18,K,J)=PRDATA(29,K,J)-DCLVL-PDSTAL
            OUTDTA(19,K,J)=PRDATA(28,K,J)-DCLVL-PDSTAL
            OUTDTA(20,K,J)=PRDATA(14,K,J)-DCLVL-PDSTAL
            OUTDTA(21,K,J)=PRDATA(7,K,J)-DCLVL-PDSTAL
            OUTDTA(22,K,J)=PRDATA(17,K,J)-DCLVL-PDSTAL
            OUTDTA(23,K,J)=PRDATA(26,K,J)-DCLVL-PDSTAL
            OUTDTA(24,K,J)=PRDATA(13,K,J)-DCLVL-PDSTAL
            OUTDTA(25,K,J)=PRDATA(20,K,J)-DCLVL-PDSTAL
            OUTDTA(26,K,J)=PRDATA(10,K,J)-DCLVL-PDSTAL
            OUTDTA(27,K,J)=PRDATA(5,K,J)-DCLVL-PDSTAL
            OUTDTA(28,K,J)=PRDATA(16,K,J)-DCLVL-PDSTAL
```

```
            OUTDTA(29,K,J)=PRDATA(8,K,J)-DCLVL-PDSTAL
            OUTDTA(30,K,J)=PRDATA(4,K,J)-DCLVL-PDSTAL
            OUTDTA(31,K,J)=PRDATA(2,K,J)-DCLVL-PDSTAL
330     CONTINUE
340     CONTINUE

**************FIND MAGNITUDE AND PHASE ***************
        DO 410 I=1,31
            DO 400 K=1,4
                MAG(I,K)=INT(SQRT(REAL(OUTDTA(I,K,1)**2+
        Z                   OUTDTA(I,K,2)**2)))/32
                IF(REAL(OUTDTA(I,K,1)).EQ.0.0)THEN
                    PHASE(I,K)=0
                ELSE
                    PHASE(I,K)=INT((ATAN2(REAL(OUTDTA(I,K,2)),
        Z                       REAL(OUTDTA(I,K,1)))*1000)/(ATAN(1)*4))
                ENDIF
                IJK=4*(I-1)+K
                MMAG(IJK)=MAG(I,K)
                PPHASE(IJK)=PHASE(I,K)
400         CONTINUE
410     CONTINUE

***************** RETURN *********************************
            RETURN
        end

************** COHERENT AVERAGING SUBROUTINE **************
        SUBROUTINE AVG(MAG,PHASE,AMAG,APHASE)
*       THIS ROUTINE TAKES MAGNITUDE AND PHASE FOR 31 SECONDS
*       OF DATA(16 CYCLES) AND COHERENTLY AVERAGES IT TO ONE
*       CYCLE, RETURNING PHASE AND MAGNITUDE.BOB DEES 28MAR89

        INTEGER*2 MAG(1984),PHASE(1984),AMAG(124),APHASE(124)
        INTEGER*2 IDTA(124),QDTA(124)

*********** CONVERT TO I & Q COMPONENTS AND SUM **************

        DO 100 I=1,124
            AMAG(I)=0
            IDTA(I)=0
            QDTA(I)=0
100     CONTINUE
        DO 200 J=1,16
            DO 150  I=1,124
            IDTA(I)=IDTA(I)+MAG((J-1)*124+I)*COS(REAL(PHASE((J-1)*124+I))
      Z             *3.141593E-3)
            QDTA(I)=QDTA(I)+MAG((J-1)*124+I)*SIN(REAL(PHASE((J-
      Z             1)*124+I))*3.141593E-3)
150         CONTINUE
200     CONTINUE
```

```
********** CONVERT TO MAG,PHASE AND RETURN **********
      DO 300 I=1,124
            AMAG(I)=INT(SQRT(IDTA(I)**2+QDTA(I)**2))
            IF(REAL(IDTA(I)).EQ.0.0)THEN
                  APHASE(I)=0
            ELSE
                  APHASE(I)=INT((ATAN2(REAL(QDTA(I)),
     Z                  REAL(IDTA(I)))*1000)/(ATAN(1)*4))
            ENDIF
300   CONTINUE
      RETURN
      END
```

B. PROGRAM AINPUT

This program is this the developement program used for testing the data input. It inputs data in the same way as AMORE but no code correlation is done. The in-phase and quadrature data are interleaved and written to disk. Also, the time is periodically written to the data file. The code correlation for this data can be done using the Fast Hadamard Transform in program AHAD. If desired, this data can be processed using other methods such as DFT or FFT.

```
*  DATA INPUT FOR FOUR CHANNELS WITH 64 HZ EXTERNAL TRIGGER
*     DATA IS INITIALLY TRANSFERRED TO MEMORY BY DMA THEN
*     BROUGHT TO THE FORTRAN PROGRAM FOR PACKAGING AND
*     WRITING TO THE BERNOULLI BOX.
*     Microsoft FORTRAN77
*     BOB DEES  3 FEB 89

      INTEGER*2 PARAM(10), DATA(7680), CHAN1(3840), CHAN2(3840)

      INTEGER*2 BASE, INTLEV, DMALEV

      INTEGER*2 MODE, RCODE, HOUR, MINUTE, CNVTIM, TIME

      INTEGER*2 DASH16, SEGADR, OFFADR, SEGPTR, OFFPTR

      INTEGER*2 TRIG, RCYC, NOC, NOS, DISP, INCR

      INTEGER*2 I, J, K, N

      INTEGER*2 SCH, FCH

      INTEGER*4 BUFFER, ALLOC

      CHARACTER*16 NAME1$, NAME2$

      CHARACTER*8 CHAN1$, CHAN2$, DATE$, STATN1$, STATN2$

115   FORMAT (' Invalid Input.  Please, try again !')

120   FORMAT (' Mode ', I2, ' ,  Error = ', I4)

124   FORMAT (' Cannot Allocate Buffer')

129   FORMAT (A16)

131   FORMAT ('POSIT=',A10,' STATION=',A10,'DATE=',A10)
```

```
132    FORMAT (A8)

133    FORMAT (I6)

134    FORMAT (2I3)

135    FORMAT ('TIME IS: HOUR= ',I3,'  MINUTE=',I3)

136    FORMAT (1X,'FINISHED WRITING TOP TO DISK, TIME=',
      Z   I3,' HR',I3,' MIN')

137    FORMAT (1X,'FINISHED WRITING BOTTOM TO DISK, TIME=',
      Z   I3,' HR',I3,' MIN 30 SEC')

171    FORMAT(' CONVERSION # =',I6)

180    FORMAT(I5)

*  INPUT INFORMATION ABOUT SIGNALS

       WRITE(*,*)'WHAT POSITION FOR CHANNEL 1 ?'
       WRITE(*,*)'EXAMPLE: L'

       READ(*,132)STATN1$

       WRITE(*,*)'CHANNEL 1 RADIO TRANSMITTER CHANNEL ?'
       WRITE(*,*)'EXAMPLE: 29'

       READ(*,132)CHAN1$

       WRITE(*,*)'WHAT POSITION FOR CHANNEL 2 ?'
       READ(*,132)STATN2$

       WRITE(*,*)'CHANNEL 2 RADIO TRANSMITTER CHANNEL ?'

       READ(*,132)CHAN2$

       WRITE(*,*)'WHAT IS THE DATE ?'
       WRITE(*,*)'EXAMPLE: 12DEC88'
       READ(*,132)DATE$
       WRITE(*,*)'BEGINNING TIME (HOUR,MINUTE)'
       WRITE(*,*)'MUST BE INTEGERS, EXAMPLE: 18,59'

       READ(*,134)HOUR,MINUTE

       WRITE(*,*)'FILENAME FOR CHANNEL 1 ?'
       WRITE(*,*)'FORMAT FOR FILENAME IS C:L1218.DAT. L FOR POSIT,'
       WRITE(*,*)'12 FOR 12DEC88, 18 FOR BEGINNING TIME 18TH HOUR, '
       WRITE(*,*)'DAT FOR SAMPLED DATA, C IS THE HARD DRIVE.'
       WRITE(*,*)'  EXAMPLE: C:L1218.DAT'
```

```
      READ(*,129)NAME1$

      WRITE(*,*)'FILENAME FOR CHANNEL 2 ?'

      READ(*,129)NAME2$

      WRITE(*,*)'HOW MANY MINUTES OF CONVERSION TIME ?'
      WRITE(*,*)'TEN MINUTES = 1.1 MBYTES'

      READ(*,133)CNVTIM
      TIME=0

      OPEN(33,FILE=NAME1$)
      OPEN(34,FILE=NAME2$)
            rewind 33
            rewind 34

            WRITE(33,131)STATN1$,CHAN1$,DATE$
            WRITE(34,131)STATN2$,CHAN2$,DATE$
            WRITE(33,135)HOUR,MINUTE
            WRITE(34,135)HOUR,MINUTE

*   Use mode 0 to initialize DASH-16 Board

      MODE = 0
      PARAM(1) = #300
      PARAM(2) = 2
      PARAM(3) = 3
      RCODE = DASH16(MODE, PARAM)
      IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE

*   FIX MEMORY BUFFER

      BUFFER = ALLOC(15360)
*   15360=64HZ*60SEC/MIN*1 MINUTE*4CHANNELS
      IF (BUFFER .EQ. 0) WRITE(*, 124)

*   EXTERNAL TRIGGER

      TRIG=0

*   REUSE SAME MEMORY AREA

      RCYC=0

*   SET A/D CHANNEL SCAN LIMITS(0-3)

      SCH=0
      FCH=3
      N = FCH - SCH + 1
```

```
*       Set Number of CONVERSIONS(30 SEC)

        NOC = 7680

600     CONTINUE

*       Use mode 1 to load Queue

        MODE = 1
        PARAM(1) = SCH
        PARAM(2) = FCH
        RCODE = DASH16(MODE, PARAM)

        IF (RCODE .NE. 0) WRITE(*, 120)MODE, RCODE

*       Perform 30 SEC OF CONVERSIONS FOR INPUTS 0-3
*   DMA TO BOTTOM OF BUFFER

610     CONTINUE

        MODE = 20
        PARAM(1) = NOC
        PARAM(2) = SEGPTR(BUFFER)
        PARAM(3) = TRIG
        PARAM(4) = RCYC
        RCODE = DASH16(MODE, PARAM)

        IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE
         IF(TIME.EQ.0)GOTO 650

*   DATA TRANSFER, MODE 9

        WRITE(*,*)'DONE WITH DMA TO TOP'
        MODE=9
        PARAM(1)=7680
        PARAM(2)=SEGPTR(BUFFER)+7680
        PARAM(3)=0
        PARAM(4)=OFFADR(DATA)
        PARAM(5)=0
        PARAM(6)=0
        RCODE=DASH16(MODE,PARAM)
        IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
        DO 620 I=1,3839,2
        CHAN1(I)=DATA(2*I-1)
        CHAN1(I+1)=DATA(2*I)
        CHAN2(I)=DATA(2*I+1)
        CHAN2(I+1)=DATA(2*I+2)
620     CONTINUE
        WRITE(33,180)CHAN1
        WRITE(34,180)CHAN2
        WRITE(33,135)HOUR,MINUTE
```

```
      WRITE(34,135)HOUR,MINUTE
      WRITE(*,136)HOUR,MINUTE
      WRITE(*,*)'  ELAPSED TIME',TIME,'  MINUTES'
      IF(TIME.GE.CNVTIM)GO TO 999

*       use mode 8 to obtain status of CONVERSION
*  RETURNS OPERATION, STATUS, CURRENT WORD COUNT
*  AS PARAMETERS

650   CONTINUE

      MODE = 8
      PARAM(3) = 0

700   CONTINUE

      RCODE = DASH16(MODE, PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
      IF(PARAM(3).LE.7680)GOTO 700

*  HALT DMA, MODE 7

      MODE=7
      RCODE=DASH16(MODE,PARAM)
      IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE

*  RESTART CONVERSIONS , DMA TO TOP OF BUFFER

      MODE = 20
      PARAM(1) = NOC
      PARAM(2) = SEGPTR(BUFFER)+7680
      PARAM(3) = TRIG
      PARAM(4) = RCYC
      RCODE = DASH16(  MODE, PARAM)

      IF(RCODE.NE.0)WRITE(*,120)MODE,RCODE
      Write(*,*)'DONE WITH DMA TO BOTTOM'

*       DATA TRANSFER, MODE 9
*  PARAMETERS 1-# OF WORDS TO TRANSFER, 2-SOURCE SEGMENT
*  3-STARTING CONVERSION NUMBER, 4-DATA ARRAY, 5-CHANNEL ARRAY

      MODE=9
      PARAM(1)=7680
      PARAM(2)=SEGPTR(BUFFER)
      PARAM(3)=0
      PARAM(4)=OFFADR(DATA)
      PARAM(5)=0
      PARAM(6)=0
      RCODE=DASH16(MODE,PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
```

```
      DO 800 I=1,3839,2
      CHAN1(I)=DATA(2*I-1)
      CHAN1(I+1)=DATA(2*I)
      CHAN2(I)=DATA(2*I+1)
      CHAN2(I+1)=DATA(2*I+2)
800   CONTINUE
      WRITE(33,180)CHAN1
      WRITE(34,180)CHAN2
      WRITE(33,135)HOUR,MINUTE
      WRITE(34,135)HOUR,MINUTE
      WRITE(*,137)HOUR,MINUTE
      WRITE(*,*)'  TIME ELAPSED',TIME,'MINUTES   30 SECONDS'

*  USE MODE 8 TO GET STATUS OF CONVERSION

      MODE = 8
      PARAM(3) = 0
900   CONTINUE
      RCODE = DASH16(MODE, PARAM)
      IF(RCODE.NE.0) WRITE(*,120)MODE,RCODE
      IF(PARAM(3).LE.7680)GOTO 900

*  HALT DMA, MODE 7

      MODE=7
      RCODE=DASH16(MODE,PARAM)
      IF (RCODE .NE. 0) WRITE(*, 120) MODE, RCODE
      TIME=TIME+1
      MINUTE=MINUTE + 1
      IF(MINUTE.GE.60)THEN
            MINUTE=MINUTE-60
            HOUR=HOUR+1
            IF(HOUR.EQ.24)HOUR=0
      ENDIF
      GOTO610
999   CONTINUE
      WRITE(*,*)'  DONE!!!',CHAR(7),CHAR(7),CHAR(7)
      STOP
      END
```

C. PROGRAM AHAD

This program performs the maximal-length sequence correlation using the Fast Hadamard Transform on the data output from the program AINPUT. It can also perform the correlation for the "block" shape of the arrival. The output of this program is of the same format as the output from AMORE.

```
*THIS PROGRAM CALLS DATA FROM ONE FILE, PERMUTES THE ORDER,
*CONDUCTS A FAST HADAMARD TRANSFORM, PERMUTES THE DATA AGAIN,
*THEN STORES THE DATA IN ANOTHER FILE. THE FIRST FILE CONTAINS
*DATA AS IN-PHASE AND QUADRATURE BASEBAND SIGNAL SAMPLES -
*COS(THETA) AND SIN(THETA). THE OUTPUT FILE WILL CONSIST OF THE
*MAGNITUDE AND PHASE OF THE CORRELATED SIGNAL. AN ADDITIONAL
*CORRELATION TO FIND THE FOUR POINT ARRIVAL CAN BE PERFORMED.


      REAL A
      INTEGER I,J,K,L,N,M,ISPACE,IWIDTH,ITOP,IBOT,TEMP
      INTEGER NI,NJ,NK
      INTEGER DX,DY,II,KK
      INTEGER*4 DCLVL,PDSTAL
      INTEGER*4 DATA(31,4,2),PRDATA(0:31,4,2),OUTDTA(31,4,2)
      INTEGER*4 MAG(31,4),PHASE(31,4),SUM(4),HOUR,MINUTE,COUNT
      CHARACTER*16 IFILE$,OFILE$
      CHARACTER*8 STATN$,CHAN$,DATE$
      CHARACTER*60 WORD$
      CHARACTER*1 DOT(62,20),ANS$

20    FORMAT (I5)
30    FORMAT (2I5)
40    FORMAT (A16)
50    FORMAT (A60)
70    FORMAT (A1)

      WRITE(*,*)'INPUT THE FILE NAME FOR THE INPUT DATA:'
      WRITE(*,*)'EXAMPLE: A:TEST1.DAT'
      READ(*,40)IFILE$
      WRITE(*,*)'INPUT THE FILE NAME FOR THE OUTPUT FILE:'
      READ(*,40)OFILE$
      WRITE(*,*)'DO YOU WANT TO DO THE SECOND CORRELATION?(Y,N)'
      READ(*,70)ANS$
      OPEN(33,FILE=IFILE$,STATUS='OLD')
      OPEN(34,FILE=OFILE$)
      REWIND 34
      READ(33,50)WORD$
```

```
      WRITE(34,*)word$
      WRITE(*,*)WORD$
      READ(33,50)WORD$
      WRITE(34,*)WORD$
      WRITE(*,*)WORD$
      COUNT=0
90    CONTINUE
      NI=0
      NK=0

********************* READ DATA ***************************
      DO 120 I=1,31
            DO 110 K=1,4
                  DO 100 J=1,2
                  COUNT=COUNT+1
                        IF (COUNT.EQ.3841)THEN
                              COUNT=1
                              NI=I
                              NK=K
                              DO 92 DX=1,62
                                    DO 91 DY=1,20
                                    DOT(DX,DY)=' '
91                                  CONTINUE
92                            CONTINUE
                              DO 94 II=1,31
                                    DO 93 KK=1,2
                                    DX=II*2+KK-2
                                    DY=MAG(II,KK)/50
                                    IF(DY.GT.20)DY=20
                                    IF(DY.LT.1)DY=1
                                    DOT(DX,DY)='*'
93                                  CONTINUE
94                            CONTINUE
                              DO 95 DY=20,1,-1
                              WRITE(*,*)(DOT(DX,DY),DX=1,62)
95                            CONTINUE


                        ENDIF
                  READ(33,20,END=999)DATA(I,K,J)
100               CONTINUE
110         CONTINUE
120   CONTINUE

****************** PERMUTE ************************
      DO 190 K=1,4
      DO 180 J=1,2
            PRDATA(0,K,J)=0
            PRDATA(10,K,J)=DATA(1,K,J)
            PRDATA(5,K,J)=DATA(2,K,J)
            PRDATA(2,K,J)=DATA(3,K,J)
```

```
            PRDATA(1,K,J)=DATA(4,K,J)
            PRDATA(16,K,J)=DATA(5,K,J)
            PRDATA(8,K,J)=DATA(6,K,J)
            PRDATA(4,K,J)=DATA(7,K,J)
            PRDATA(18,K,J)=DATA(8,K,J)
            PRDATA(9,K,J)=DATA(9,K,J)
            PRDATA(20,K,J)=DATA(10,K,J)
            PRDATA(26,K,J)=DATA(11,K,J)
            PRDATA(13,K,J)=DATA(12,K,J)
            PRDATA(6,K,J)=DATA(13,K,J)
            PRDATA(19,K,J)=DATA(14,K,J)
            PRDATA(25,K,J)=DATA(15,K,J)
            PRDATA(28,K,J)=DATA(16,K,J)
            PRDATA(30,K,J)=DATA(17,K,J)
            PRDATA(31,K,J)=DATA(18,K,J)
            PRDATA(15,K,J)=DATA(19,K,J)
            PRDATA(7,K,J)=DATA(20,K,J)
            PRDATA(3,K,J)=DATA(21,K,J)
            PRDATA(17,K,J)=DATA(22,K,J)
            PRDATA(24,K,J)=DATA(23,K,J)
            PRDATA(12,K,J)=DATA(24,K,J)
            PRDATA(22,K,J)=DATA(25,K,J)
            PRDATA(27,K,J)=DATA(26,K,J)
            PRDATA(29,K,J)=DATA(27,K,J)
            PRDATA(14,K,J)=DATA(28,K,J)
            PRDATA(23,K,J)=DATA(29,K,J)
            PRDATA(11,K,J)=DATA(30,K,J)
            PRDATA(21,K,J)=DATA(31,K,J)
180     CONTINUE
190     CONTINUE
****************** FAST HADAMARD*****************

      DO 300 K=1,4
      DO 290 J=1,2
            DO 270 L=1,5
                  ISPACE=2**L
                  IWIDTH=2**(L-1)
                  DO 250 N=0,(IWIDTH-1)
                        DO 230 ITOP=N,(32-2),ISPACE
                              IBOT=ITOP+IWIDTH
                              TEMP=PRDATA(IBOT,K,J)
                              PRDATA(IBOT,K,J)=PRDATA(ITOP,K,J)-TEMP
                              PRDATA(ITOP,K,J)=PRDATA(ITOP,K,J)+TEMP
230                     CONTINUE
250                 CONTINUE
270         CONTINUE
290     CONTINUE
300     CONTINUE
************ PERMUTE AND REMOVE BIAS ***************

      DO 340 K=1,4
```

```
      DO 330 J=1,2
            DCLVL=(ABS(DATA(1,K,J))-PRDATA(0,K,J))/30
            PDSTAL=PRDATA(0,K,J)+DCLVL*31
* NOTE:DCLVL ISN'T THE SAME AS THE PEDESTAL
            OUTDTA(1,K,J)=PRDATA(1,K,J)-DCLVL-PDSTAL
            OUTDTA(2,K,J)=PRDATA(18,K,J)-DCLVL-PDSTAL
            OUTDTA(3,K,J)=PRDATA(9,K,J)-DCLVL-PDSTAL
            OUTDTA(4,K,J)=PRDATA(22,K,J)-DCLVL-PDSTAL
            OUTDTA(5,K,J)=PRDATA(11,K,J)-DCLVL-PDSTAL
            OUTDTA(6,K,J)=PRDATA(23,K,J)-DCLVL-PDSTAL
            OUTDTA(7,K,J)=PRDATA(25,K,J)-DCLVL-PDSTAL
            OUTDTA(8,K,J)=PRDATA(30,K,J)-DCLVL-PDSTAL
            OUTDTA(9,K,J)=PRDATA(15,K,J)-DCLVL-PDSTAL
            OUTDTA(10,K,J)=PRDATA(21,K,J)-DCLVL-PDSTAL
            OUTDTA(11,K,J)=PRDATA(24,K,J)-DCLVL-PDSTAL
            OUTDTA(12,K,J)=PRDATA(12,K,J)-DCLVL-PDSTAL
            OUTDTA(13,K,J)=PRDATA(6,K,J)-DCLVL-PDSTAL
            OUTDTA(14,K,J)=PRDATA(3,K,J)-DCLVL-PDSTAL
            OUTDTA(15,K,J)=PRDATA(19,K,J)-DCLVL-PDSTAL
            OUTDTA(16,K,J)=PRDATA(27,K,J)-DCLVL-PDSTAL
            OUTDTA(17,K,J)=PRDATA(31,K,J)-DCLVL-PDSTAL
            OUTDTA(18,K,J)=PRDATA(29,K,J)-DCLVL-PDSTAL
            OUTDTA(19,K,J)=PRDATA(28,K,J)-DCLVL-PDSTAL
            OUTDTA(20,K,J)=PRDATA(14,K,J)-DCLVL-PDSTAL
            OUTDTA(21,K,J)=PRDATA(7,K,J)-DCLVL-PDSTAL
            OUTDTA(22,K,J)=PRDATA(17,K,J)-DCLVL-PDSTAL
            OUTDTA(23,K,J)=PRDATA(26,K,J)-DCLVL-PDSTAL
            OUTDTA(24,K,J)=PRDATA(13,K,J)-DCLVL-PDSTAL
            OUTDTA(25,K,J)=PRDATA(20,K,J)-DCLVL-PDSTAL
            OUTDTA(26,K,J)=PRDATA(10,K,J)-DCLVL-PDSTAL
            OUTDTA(27,K,J)=PRDATA(5,K,J)-DCLVL-PDSTAL
            OUTDTA(28,K,J)=PRDATA(16,K,J)-DCLVL-PDSTAL
            OUTDTA(29,K,J)=PRDATA(8,K,J)-DCLVL-PDSTAL
            OUTDTA(30,K,J)=PRDATA(4,K,J)-DCLVL-PDSTAL
            OUTDTA(31,K,J)=PRDATA(2,K,J)-DCLVL-PDSTAL
330   CONTINUE
340   CONTINUE

************** 4 POINT CORRELATION ********************

      IF(ANS$.EQ.'N'.OR.ANS$.EQ.'n')GOTO395
      write(*,*)'square correlate'
      DO 390 I=1,30
            DO 380 K=1,4
            IF(K.EQ.1)THEN
                  K2=2
                  K3=3
                  K4=4
                  I2=I
                  I3=I
                  I4=I
```

```
            ELSEIF(K.EQ.2)THEN
                  K2=3
                  K3=4
                  K4=1
                  I2=I
                  I3=I
                  I4=I+1
            ELSE IF(K.EQ.3)THEN
                  K2=4
                  K3=1
                  K4=2
                  I2=I
                  I3=I+1
                  I4=I+1
            ELSE IF(K.EQ.4)THEN
                  K2=1
                  K3=2
                  K4=3
                  I2=I+1
                  I3=I+1
                  I4=I+1
            ENDIF
            DO 370 J=1,2
             OUTDTA(I,K,J)=OUTDTA(I,K,J)+OUTDTA(I2,K2,J)
      OUTDTA(I,K,J)=OUTDTA(I,K,J)+OUTDTA(I3,K3,J)+OUTDTA(I4,K4,J)
       OUTDTA(I,K,J)=OUTDTA(I,K,J)/4
370         CONTINUE
380         CONTINUE
390   CONTINUE
395   CONTINUE
*************FIND MAGNITUDE AND PHASE ***************
      DO 410 I=1,31
            DO 400 K=1,4
                  MAG(I,K)=INT(SQRT(REAL(OUTDTA(I,K,1)**2+
      Z                 OUTDTA(I,K,2)**2)))/32
                  IF(REAL(OUTDTA(I,K,1)).EQ.0.0)THEN
                        PHASE(I,K)=1571
                  ELSE
                        PHASE(I,K)=INT((ATAN2(REAL(OUTDTA(I,K,2)),
      Z                       REAL(OUTDTA(I,K,1)))*1000)/(ATAN(1)*4))
                  ENDIF
400         CONTINUE
410   CONTINUE

***************** WRITE TO DISK *********************************

      DO 460 I=1,31
            DO 450 K=1,4
*                 IF(NI.EQ.I.AND.NK.EQ.K)WRITE(34,50)WORD$
                  WRITE(34,30)MAG(I,K),PHASE(I,K)
450         CONTINUE
```

```
460     CONTINUE
*       IF(COUNT.LT.300)READ(*,*) A
        IF(A.NE.999)GO TO 90

********************** END *****************

999     WRITE(*,*)'FINISHED',CHAR(7),CHAR(7),CHAR(7)
        stop
        end
```

D. PROGRAM ACRID

This program inputs a file of magnitude and phase measurements produced by AHAD or AMORE and provides several post-processing options. The data sequences can be coherently averaged for 2,4,8,or 16 sequences. Coherent averaging converts the magnitude and phase to in-phase and quadrature components and, taking the selected number of sequences, finds the mean of the components. If the sequences are correlated and the noise is not, this results in an increase in the SNR. In this case the rate at which the travel time is changing determines the amount of increase possible. The coherent average lowers the sampling rate of the ocean data and so is not necessarily desirable. The next option in processing the signal is to correlate for a block the width of the signal. This is a discrete shape correlation, not correlating for amplitude. The result is an increase in the sharpness of the peak and a reduction in high frequency noise. At this point the data is converted again to magnitude and phase and can be written to a disk file. The data can be "peak-picked." A fixed window can be set where the program will perform a cubic spline curve fit to the data, generate floating point real numbers for points every .9765 milliseconds. The program "picks the peak" value as the arrival time estimate and writes the clock time, arrival time estimate, magnitude and signal-to-noise ratio to a file. The user inputs a threshold for the SNR on the peak-picking. If the signal does not meet this threshold, the last peak is repeated with the low SNR recorded. This gives evenly spaced data for FFT periodogram and power spectrum analysis.

```
*     PROGRAM FOR POST PROCESSING AND INTERACTIVE PEAK PICKING
*     BOB DEES 13APR89

      CHARACTER*20 IFILE$,OFILE$,PFILE$
      CHARACTER*1 ANS1$,ANS2$,ANS3$,ANS4$
      CHARACTER*60 HEADR$
      CHARACTER*15 H1$,H2$
      INTEGER*2 MAG(3968),PHASE(3968),CDTA(3968),SDTA(3968)
      INTEGER*2 ACDTA(3968),ASDTA(3968),SUMC(3),SUMCA(3)
      INTEGER*2 SUMS(3),SUMSA(3),PEAK,IPEAK,PPHASE,HPEAK
      INTEGER*2 RANGE,LOW,HI,IMAG(124),PMAG(23)
      INTEGER*4 NOISE
      INTEGER I,J,K,N,NUM,HOUR,MINUTE
```

```
      REAL SNR,THRESH,SECOND,DELTAT,MAG16(-160:161),MAXMAG
      REAL HMXMAG

10    FORMAT(A20)
20    FORMAT(A1)
30    FORMAT(A60)
40    FORMAT(A19,A3,A1,I2,A19,A3)
50    FORMAT(2I5)
60    FORMAT('TIME=  ',I2,':',I2,':',F7.3,'  PEAK=',I4,
     Z '  MAG=',F9.3,'  SNR=',F6.3)
61    FORMAT(1X,I2,'  TIME=  ',I2,':',I2,':',F7.3,'   PEAK=',I4,
     Z '  MAG=',F9.3,'  SNR=',F6.3)
62    FORMAT(1X,I2,'  TIME=  ',I2,':',I2,':',F7.3,'   PEAK=',I4,
     Z '  MAG=',F9.3,'  SNR=',F6.3,'BELOW THRESHOLD')
70    FORMAT(A15,I3,A9,I3)
71    FORMAT(1X,A15,I3,A9,I3)

      SUMC(1)=0
      SUMC(2)=0
      SUMC(3)=0
      SUMS(1)=0
      SUMS(2)=0
      SUMS(3)=0
      SECOND=0.0

************** INPUT FILE INFO **********************

100   WRITE(*,*)'POST PROCESSING OF THE DATA IS BY EITHER'
      WRITE(*,*)'COHERENT AVERAGING, CORRELATION FOR THE EXPECTED'
      WRITE(*,*)'"BLOCK" ARRIVAL, OR BOTH.'
      WRITE(*,*)'CHOICES AVAILABLE FOR THE COHERENT ADD ARE A '
      WRITE(*,*)'SUM OF 1,2,4,8, OR 16 SERIES.'
      WRITE(*,*)'THIS PROGRAM WILL READ THE DATA FILE UNTIL IT'
      WRITE(*,*)'REACHES AN "END OF FILE". DATA WILL BE WRITTEN'
      WRITE(*,*)'TO THE OUTPUT FILE IN THE SAME FORMAT AS THE'
      WRITE(*,*)'INPUT FILE.'
      WRITE(*,*)' '
      WRITE(*,*)'WHAT IS THE INPUT FILENAME?'
      READ(*,10)IFILE$
      OPEN(33,FILE=IFILE$,STATUS='OLD')
      WRITE(*,*)'DO YOU WANT TO WRITE THE OUTPUT DATA TO A FILE?'
      READ(*,20)ANS3$
      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')THEN
            WRITE(*,*)'WHAT IS THE OUTPUT FILENAME?'
            READ(*,10)OFILE$
            OPEN(34,FILE=OFILE$)
            REWIND 34
      ENDIF
      WRITE(*,*)'DO YOU WANT TO DO COHERENT AVERAGING?'
      READ(*,20)ANS1$
      IF(ANS1$.EQ.'Y'.OR.ANS1$.EQ.'y')THEN
```

```
            WRITE(*,*)'HOW MANY?'
            READ(*,*)NUM
            IF(NUM.NE.1.AND.NUM.NE.2.AND.NUM.NE.4.AND.NUM.NE.8.AND.
     Z NUM.NE.16)GOTO 100
      ELSE
            NUM=1
      ENDIF
      DELTAT=REAL(NUM)*1.9375
      WRITE(*,*)'DO YOU WANT TO DO THE "BLOCK" CORRELATION?'
      READ(*,20)ANS2$
      WRITE(*,*)'DO YOU WANT TO PEAK-PICK?'
      READ(*,20)ANS4$
      IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')THEN
            WRITE(*,*)'WHAT OUTPUT FILE DOES THIS GO TO?'
            READ(*,10)PFILE$
            OPEN(35,FILE=PFILE$)
            REWIND 35
            WRITE(*,*)'WHAT INITIAL POINT FOR THE PEAK(1-124)'
            READ(*,*)IPEAK
            WRITE(*,*)'THEN WHAT RANGE TO SEARCH?'
            READ(*,*)RANGE
            WRITE(*,*)'INPUT A SIGNAL TO NOISE RATIO THRESHOLD:'
            READ(*,*)THRESH
            LOW=IPEAK-RANGE
            HI=IPEAK+RANGE
      ENDIF

      READ(33,30)HEADR$
      WRITE(*,*)HEADR$
      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')WRITE(34,30)HEADR$
      IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')WRITE(35,30)HEADR$
      READ(33,70)H1$,HOUR,H2$,MINUTE
      WRITE(*,71)H1$,HOUR,H2$,MINUTE
      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')WRITE(34,71)H1$,HOUR,H2$,MINUTE
      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')WRITE(34,40)'COHERENT AVERAGE:',
     Z      ANS1$,'x',NUM,'  BLOCK CORRELATION:',ANS2$
      IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')WRITE(35,71)H1$,HOUR,H2$,MINUTE

      IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')WRITE(35,40)'COHERENT AVERAGE:',
     Z      ANS1$,'x',NUM,'  BLOCK CORRELATION:',ANS2$

********** READ DATA **********

105   CONTINUE
      DO 110 N=1,3968
      READ(33,50,END=999)MAG(N),PHASE(N)
      CDTA(N)=MAG(N)*COS(REAL(PHASE(N))*3.141593E-3)
      SDTA(N)=MAG(N)*SIN(REAL(PHASE(N))*3.141593E-3)
110   CONTINUE

********** COHERENT AVERAGING **********
```

```
      DO 170 I=1,3968
            ACDTA(I)=0
            ASDTA(I)=0
170   CONTINUE
      DO 200 I=0,((32/NUM)-1)
            DO 190 J=0,(NUM-1)
                  DO 180 K=1,124

      ACDTA(I*124+K)=ACDTA(I*124+K)+CDTA(K+(J*124)+(I*NUM*124))

      ASDTA(I*124+K)=ASDTA(I*124+K)+SDTA(K+(J*124)+(I*NUM*124))
180               CONTINUE
190         CONTINUE
            DO 195 K=1,124
                  ACDTA(I*124+K)=ACDTA(I*124+K)/NUM
                  ASDTA(I*124+K)=ASDTA(I*124+K)/NUM
195         CONTINUE
200   CONTINUE

****************** BLOCK CORRELATION ***************

      IF(ANS2$.NE.'Y'.AND.ANS2$.NE.'y')GOTO 390
            SUMCA(1)=ACDTA(3968/NUM-2)+ACDTA(3968/NUM-
     Z 1)+ACDTA(3968/NUM)
            SUMCA(2)=ACDTA(3968/NUM-1)+ACDTA(3968/NUM)
            SUMCA(3)=ACDTA(3968/NUM)
            SUMSA(1)=ASDTA(3968/NUM-2)+ASDTA(3968/NUM-
     Z 1)+ASDTA(3968/NUM)
            SUMSA(2)=ASDTA(3968/NUM-1)+ASDTA(3968/NUM)
            SUMSA(3)=ASDTA(3968/NUM)
      DO 300 I=(3968/NUM),4,-1
            ACDTA(I)=ACDTA(I)+ACDTA(I-1)+ACDTA(I-2)+ACDTA(I-3)
            ASDTA(I)=ASDTA(I)+ASDTA(I-1)+ASDTA(I-2)+ASDTA(I-3)
300   CONTINUE
      ACDTA(3)=ACDTA(3)+ACDTA(2)+ACDTA(1)+SUMC(3)
      ACDTA(2)=ACDTA(2)+ACDTA(1)+SUMC(2)
      ACDTA(1)=ACDTA(1)+SUMC(1)
      SUMC(1)=SUMCA(1)
      SUMC(2)=SUMCA(2)
      SUMC(3)=SUMCA(3)
      ASDTA(3)=ASDTA(3)+ASDTA(2)+ASDTA(1)+SUMS(3)
      ASDTA(2)=ASDTA(2)+ASDTA(1)+SUMS(2)
      ASDTA(1)=ASDTA(1)+SUMS(1)
      SUMS(1)=SUMSA(1)
      SUMS(2)=SUMSA(2)
      SUMS(3)=SUMSA(3)

*************** CONVERT TO MAGNITUDE AND PHASE ***************

390   CONTINUE
      DO 400 I=1,(3968/NUM)
```

```
      MAG(I)=INT(SQRT(REAL(ACDTA(I))**2+REAL(ASDTA(I))**2))
      IF(REAL(ACDTA(I)).EQ.0.)THEN
            PHASE(I)=0
      ELSE
            PHASE(I)=INT((ATAN2(REAL(ASDTA(I)),REAL(ACDTA(I)))*1000)/
     Z (ATAN(1)*4))
      ENDIF
400   CONTINUE

***************** WRITE TO DISK *******************************

      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')THEN
      WRITE(34,50)(MAG(I),PHASE(I),I=1,3968/NUM)
      ENDIF

***************** PEAK PICKING ********************************
      IF(ANS4$.EQ.'n'.OR.ANS4$.EQ.'N')GOTO 105
      DO 800 I=1,(32/NUM)
            NOISE=0
            MAXMAG=0
            DO 650 J=1,124
                  IMAG(J)=MAG((I-1)*124+J)
                  NOISE=NOISE+IMAG(J)
650         CONTINUE
            N=1
            DO 660 J=IPEAK-11,IPEAK+11
                  IF(J.LT.1)THEN
                        K=124+J
                  ELSE IF(J.GT.124)THEN
                        K=J-124
                  ELSE
                        K=J
                  ENDIF
                  PMAG(N)=IMAG(K)
                  N=N+1
660         CONTINUE

            CALL SPLINE(PMAG,MAG16)

            DO 670 J=(RANGE*(-16)),(RANGE*16)
                  IF(MAG16(J).GT.MAXMAG)THEN
                        PEAK=J
                        MAXMAG=MAG16(J)
                  ENDIF
670         CONTINUE
            PEAK=PEAK+(IPEAK-1)*16
            IF(PEAK.GT.1984)PEAK=PEAK-1984
            IF(PEAK.LT.1)PEAK=PEAK+1984
*     COUNT=COUNT+1
*     IF(COUNT.EQ.20)THEN
*           OPEN(40,FILE='c:\matlab\zp.mat')
```

```
*           OPEN(41,FILE='c:\matlab\z16.mat')
*           open(42,file='c:\matlab\zi.mat')
*           WRITE(40,*)PMAG
*           WRITE(41,*)MAG16
*           write(42,*)imag
*           WRITE(*,*)'ipeak= ',ipeak,' PEAK= ',PEAK^J*          GOTO 999
*     ENDIF

****************** TIME *****************
            SECOND=SECOND+DELTAT
            IF(SECOND.GE.60.0)THEN
                  SECOND=SECOND-60.
                  MINUTE=MINUTE+1
                  IF(MINUTE.GE.60)THEN
                        MINUTE=MINUTE-60
                        HOUR=HOUR+1
                        IF(HOUR.GE.24)HOUR=HOUR-24
                  ENDIF
            ENDIF
************** SNR ************************

            IF(NOISE.LT.124)NOISE=124
            NOISE=NOISE/124
            SNR=20*LOG10(REAL(MAXMAG)/REAL(NOISE))
            IF(SNR.LT.THRESH)THEN

      WRITE(*,62)I,HOUR,MINUTE,SECOND,PEAK,MAXMAG,SNR
                  PEAK=HPEAK
                  MAXMAG=HMXMAG
            ELSE

      WRITE(*,61)I,HOUR,MINUTE,SECOND,PEAK,MAXMAG,SNR
            ENDIF
            WRITE(35,60)HOUR,MINUTE,SECOND,PEAK,MAXMAG,SNR
            HPEAK=PEAK
            HMXMAG=MAXMAG
800   CONTINUE
      GO TO 105

********************** END *********************

999   WRITE(*,*)'FINISHED',CHAR(7),CHAR(7),CHAR(7)
      stop
      end
***************** CUBIC SPLINE INTERPOLATION **********************

      SUBROUTINE SPLINE(MAG,MAG16)
      INTEGER*2 MAG(23)
      REAL MAGD2(23),U(23),MAG16(16:337)
***** SET BOUNDARY CONDITIONS ******
      MAGD2(1)=0.
```

```
      U(1)=0.
      MAGD2(23)=0.
***** FIND SECOND DERIVATIVES ******
      DO 11 I=2,22
            P=MAGD2(I-1)/2.+2.
            MAGD2(I)=(-.5)/P
            U(I)=(3.*(MAG(I+1)-2*MAG(I)+MAG(I-1))-.5*U(I-1))/P
11    CONTINUE
      DO 12 K=22,1,-1
            MAGD2(K)=MAGD2(K)*MAGD2(K+1)+U(K)
12    CONTINUE
***** SPLINE TO INTERPOLATE ********
      DO 13 I=16,337
            J=I/16
            IF(REAL(I)/16.0.EQ.J)THEN
                  MAG16(I)=REAL(MAG(J))
            ELSE
                  A=REAL((J+1)*16-I)/16.0
                  B=REAL(I-J*16)/16.0
                  MAG16(I)=A*MAG(J)+B*MAG(J+1)+((A**3-A)*MAGD2(J)+
     Z                  (B**3-B)*MAGD2(J+1))/6.
                  ENDIF
13    CONTINUE
      RETURN
      END
```

E. PROGRAM AGONY

This program accomplishes coherent averaging and block correlation in the same way as the program ACRID. The difference is in the "peak-picking." This program finds the peak but if the SNR is below a user set threshold, the program stops, draws the current arrival period on the screen, and asks the user to find the peak. The user then resets any parameters and decides whether or not to add the chosen arrival time estimate to the output data file. Note that this can lead to uneven separation between data points, which cannot then be used in FFT power spectrum and periodogram analysis. The window where the program looks for the peak amplitude is of fixed width but resets its center to the previous arrival time estimate on each cycle.

```
*     PROGRAM FOR POST PROCESSING AND INTERACTIVE PEAK PICKING
*     BOB DEES 13APR89

      CHARACTER*20 IFILE$,OFILE$,PFILE$
      CHARACTER*1 ANS1$,ANS2$,ANS3$,ANS4$,ANS5$
      CHARACTER*60 HEADR$
      CHARACTER*15 H1$,H2$
      INTEGER*2 MAG(3968),PHASE(3968),CDTA(3968),SDTA(3968)
      INTEGER*2 ACDTA(3968),ASDTA(3968),SUMC(3),SUMCA(3)
      INTEGER*2 SUMS(3),SUMSA(3),PEAK,IPEAK,MAXMAG,PPHASE
      INTEGER*2 ANGE,LOW,HI,MAG16(1984),IMAG(0:124)
      INTEGER*4 NOISE
      INTEGER I,J,K,N,NUM,INIT,YDIV,HOUR,MINUTE
      REAL SNR,THRESH,SECOND,DELTAT

10    FORMAT(A20)
20    FORMAT(A1)
30    FORMAT(A60)
40    FORMAT(A19,A3,A1,I2,A19,A3)
50    FORMAT(2I5)
60    FORMAT('TIME=  ',I2,':',I2,':',F7.3,'  PEAK=',I4,
     Z '  MAG=',I5,'  SNR=',F6.3)
61    FORMAT(1X,I2,'  TIME=  ',I2,':',I2,':',F7.3,'   PEAK=',I4,
     Z '  MAG=',I5,'  SNR=',F6.3)
70    FORMAT(A15,I3,A9,I3)
71    FORMAT(1X,A15,I3,A9,I3)

      SUMC(1)=0
      SUMC(2)=0
      SUMC(3)=0
```

```
      SUMS(1)=0
      SUMS(2)=0
      SUMS(3)=0
      INIT=0
      IMAG(0)=0
      YDIV=100
      SECOND=0.0

************** INPUT FILE INFO **********************

100   WRITE(*,*)'POST PROCESSING OF THE DATA IS BY EITHER'
      WRITE(*,*)'COHERENT AVERAGING, CORRELATION FOR THE'
      WRITE(*,*)'EXPECTED "BLOCK" ARRIVAL, OR BOTH.'
      WRITE(*,*)'CHOICES AVAILABLE FOR THE COHERENT ADD ARE A '
      WRITE(*,*)'SUM OF 1,2,4,8, OR 16 SERIES.'
      WRITE(*,*)'THIS PROGRAM WILL READ THE DATA FILE UNTIL IT'
      WRITE(*,*)'REACHES AN "END OF FILE". DATA WILL BE WRITTEN'
      WRITE(*,*)'TO THE OUTPUT FILE IN THE SAME FORMAT AS THE'
      WRITE(*,*)'INPUT FILE.'
      WRITE(*,*)' '
      WRITE(*,*)'WHAT IS THE INPUT FILENAME?'
      READ(*,10)IFILE$
      OPEN(33,FILE=IFILE$,STATUS='OLD')
      WRITE(*,*)'DO YOU WANT TO WRITE THE OUTPUT DATA TO A FILE?'
      READ(*,20)ANS3$
      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')THEN
            WRITE(*,*)'WHAT IS THE OUTPUT FILENAME?'
            READ(*,10)OFILE$
            OPEN(34,FILE=OFILE$)
            REWIND 34
      ENDIF
      WRITE(*,*)'DO YOU WANT TO DO COHERENT AVERAGING?'
      READ(*,20)ANS1$
      IF(ANS1$.EQ.'Y'.OR.ANS1$.EQ.'y')THEN
            WRITE(*,*)'HOW MANY?'
            READ(*,*)NUM

      IF(NUM.NE.1.AND.NUM.NE.2.AND.NUM.NE.4.AND.NUM.NE.8.AND.
     Z NUM.NE.16)GOTO 100
       ELSE
            NUM=1
      ENDIF
      DELTAT=NUM*1.9375
      WRITE(*,*)'DO YOU WANT TO DO THE "BLOCK" CORRELATION?'
      READ(*,20)ANS2$
      WRITE(*,*)'DO YOU WANT TO PEAK-PICK?'
      READ(*,20)ANS4$
      IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')THEN
            WRITE(*,*)'WHAT OUTPUT FILE DOES THIS GO TO?'
            READ(*,10)PFILE$
            OPEN(35,FILE=PFILE$)
```

```
            REWIND 35
        ENDIF

        READ(33,30)HEADR$
        WRITE(*,*)HEADR$
        IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')WRITE(34,30)HEADR$
        IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')WRITE(35,30)HEADR$
        READ(33,70)H1$,HOUR,H2$,MINUTE
        WRITE(*,71)H1$,HOUR,H2$,MINUTE
        IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')WRITE(34,71)H1$,HOUR,H2$,
       Z MINUTE
        IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')WRITE(34,40)'COHERENT
       Z AVERAGE:',ANS1$,'x',NUM,'  BLOCK CORRELATION:',ANS2$
        IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')WRITE(35,71)H1$,HOUR,H2$,
       Z MINUTE

        IF(ANS4$.EQ.'Y'.OR.ANS4$.EQ.'y')WRITE(35,40)'COHERENT
       Z  AVERAGE:',ANS1$,'x',NUM,'  BLOCK CORRELATION:',ANS2$

**************** READ DATA *********************************

105     CONTINUE
        DO 110 N=1,3968
        READ(33,50,END=999)MAG(N),PHASE(N)
        CDTA(N)=MAG(N)*COS(REAL(PHASE(N))*3.141593E-3)
        SDTA(N)=MAG(N)*SIN(REAL(PHASE(N))*3.141593E-3)
110     CONTINUE

**************** COHERENT AVERAGING ********************
        DO 170 I=1,3968
                ACDTA(I)=0
                ASDTA(I)=0
170     CONTINUE
        DO 200 I=0,((32/NUM)-1)
                DO 190 J=0,(NUM-1)
                        DO 180 K=1,124

        ACDTA(I*124+K)=ACDTA(I*124+K)+CDTA(K+(J*124)+(I*NUM*124))

        ASDTA(I*124+K)=ASDTA(I*124+K)+SDTA(K+(J*124)+(I*NUM*124))
180                     CONTINUE
190             CONTINUE
                DO 195 K=1,124
                        ACDTA(I*124+K)=ACDTA(I*124+K)/NUM
                        ASDTA(I*124+K)=ASDTA(I*124+K)/NUM
195             CONTINUE
200     CONTINUE

******************** BLOCK CORRELATION ***************

        IF(ANS2$.NE.'Y'.AND.ANS2$.NE.'y')GOTO 390
```

```
            SUMCA(1)=ACDTA(3968/NUM-2)+ACDTA(3968/NUM-
     Z 1)+ACDTA(3968/NUM)
            SUMCA(2)=ACDTA(3968/NUM-1)+ACDTA(3968/NUM)
            SUMCA(3)=ACDTA(3968/NUM)
            SUMSA(1)=ASDTA(3968/NUM-2)+ASDTA(3968/NUM-
     Z 1)+ASDTA(3968/NUM)
            SUMSA(2)=ASDTA(3968/NUM-1)+ASDTA(3968/NUM)
            SUMSA(3)=ASDTA(3968/NUM)
      DO 300 I=(3968/NUM),4,-1
            ACDTA(I)=ACDTA(I)+ACDTA(I-1)+ACDTA(I-2)+ACDTA(I-3)
            ASDTA(I)=ASDTA(I)+ASDTA(I-1)+ASDTA(I-2)+ASDTA(I-3)
300   CONTINUE
      ACDTA(3)=ACDTA(3)+ACDTA(2)+ACDTA(1)+SUMC(3)
      ACDTA(2)=ACDTA(2)+ACDTA(1)+SUMC(2)
      ACDTA(1)=ACDTA(1)+SUMC(1)
      SUMC(1)=SUMCA(1)
      SUMC(2)=SUMCA(2)
      SUMC(3)=SUMCA(3)
      ASDTA(3)=ASDTA(3)+ASDTA(2)+ASDTA(1)+SUMS(3)
      ASDTA(2)=ASDTA(2)+ASDTA(1)+SUMS(2)
      ASDTA(1)=ASDTA(1)+SUMS(1)
      SUMS(1)=SUMSA(1)
      SUMS(2)=SUMSA(2)
      SUMS(3)=SUMSA(3)

*************** CONVERT TO MAGNITUDE AND PHASE ***************

390   CONTINUE
      DO 400 I=1,(3968/NUM)
      MAG(I)=INT(SQRT(REAL(ACDTA(I))**2+REAL(ASDTA(I))**2))
      IF(REAL(ACDTA(I)).EQ.0.)THEN
            PHASE(I)=0
      ELSE
            PHASE(I)=INT((ATAN2(REAL(ASDTA(I)),REAL(ACDTA(I)))*1000)/
    Z (ATAN(1)*4))
      ENDIF
400   CONTINUE

***************** WRITE TO DISK ******************************

      IF(ANS3$.EQ.'Y'.OR.ANS3$.EQ.'y')THEN
      WRITE(34,50)(MAG(I),PHASE(I),I=1,3968/NUM)
      ENDIF

***************** PEAK PICKING *******************************
      IF(ANS4$.EQ.'n'.OR.ANS4$.EQ.'N')GOTO 105
      DO 800 I=1,(32/NUM)
            DO 650 J=1,124
                  IMAG(J)=MAG((I-1)*124+J)
650         CONTINUE
      IF(INIT.EQ.0)THEN
```

```
            CALL DRAW(IMAG,YDIV)
            WRITE(*,*)'WHAT INITIAL POINT FOR THE PEAK(1-124)'
            READ(*,*)IPEAK
            IPEAK=IPEAK*16
            WRITE(*,*)'THEN WHAT RANGE TO SEARCH?'
            READ(*,*)RANGE
            RANGE=RANGE*16
            WRITE(*,*)'INPUT A SIGNAL TO NOISE RATIO THRESHOLD:'
            READ(*,*)THRESH
            INIT=1
      ENDIF

            CALL SPLINE(IMAG,MAG16)
            LOW=IPEAK-RANGE
            HI=IPEAK+RANGE
            MAXMAG=0
            NOISE=0
            IF(LOW.LT.1)THEN
                  DO 700 J=(1984+LOW),1984
                        IF(MAG16(J).GT.MAXMAG)THEN
                        MAXMAG=MAG16(J)
                        PEAK=J
                        ENDIF
700               CONTINUE
                  DO 705 J=1,HI
                        IF(MAG16(J).GT.MAXMAG)THEN
                        MAXMAG=MAG16(J)
                        PEAK=J
                        ENDIF
705               CONTINUE
            ELSE IF(HI.GT.1984)THEN
                  DO 710 J=1,(HI-1984)
                        IF(MAG16(J).GT.MAXMAG)THEN
                        MAXMAG=MAG16(J)
                        PEAK=J
                        ENDIF
710               CONTINUE
                  DO 715 J=LOW,1984
                        IF(MAG16(J).GT.MAXMAG)THEN
                        MAXMAG=MAG16(J)
                        PEAK=J
                        ENDIF
715               CONTINUE
            ELSE IF(HI.LE.1984.AND.LOW.GE.1)THEN
                  DO 720 J=LOW,HI
                        IF(MAG16(J).GT.MAXMAG)THEN
                        MAXMAG=MAG16(J)
                        PEAK=J
                        ENDIF
720               CONTINUE
            ELSE
```

```
                WRITE(*,*)'RANGE PROBLEM IN PEAK PICKING',LOW,HI
                GO TO 999
            ENDIF
            DO 725 J=1,124
            NOISE=NOISE+MAG(J+(I-1)*124)

725         CONTINUE
            IF(NOISE.LT.124)NOISE=124
            NOISE=NOISE/124
            SNR=20*LOG10(REAL(MAXMAG)/REAL(NOISE))
            IF(SNR.LT.THRESH)THEN
                WRITE(*,*)CHAR(7),CHAR(7),CHAR(7)
                CALL DRAW(IMAG,YDIV)
                WRITE(*,*)'SNR: ',SNR,'  PEAK:  ',PEAK/16
                WRITE(*,*)'DO YOU WANT TO HOLD THE PREVIOUS
     Z  PEAK?'
                READ(*,20)ANS5$
                IF(ANS5$.EQ.'N'.OR.ANS5$.EQ.'n')THEN
750                 WRITE(*,*)'SPECIFY PEAK:'
                    READ(*,*)PEAK
                    IF(PEAK.GT.124.OR.PEAK.LT.1)GOTO750
                    PEAK=PEAK*16
                ELSE
                    PEAK=IPEAK
                ENDIF
                WRITE(*,*)'DO YOU WANT TO CHANGE THE
    Z THRESHOLD?'
                READ(*,20)ANS5$
                IF(ANS5$.EQ.'Y'.OR.ANS5$.EQ.'y')THEN
                    WRITE(*,*)'SPECIFY THRESHOLD:'
                    READ(*,*)THRESH
                ENDIF
            ENDIF
            SECOND=SECOND+DELTAT
            IF(SECOND.GE.60.0)THEN
                SECOND=SECOND-60.
                MINUTE=MINUTE+1
                IF(MINUTE.GE.60)THEN
                    MINUTE=MINUTE-60
                    HOUR=HOUR+1
                    IF(HOUR.GE.24)HOUR=HOUR-24
                ENDIF
            ENDIF
            WRITE(35,60)HOUR,MINUTE,SECOND,PEAK,MAXMAG,SNR
            WRITE(*,61)I,HOUR,MINUTE,SECOND,PEAK,MAXMAG,SNR
            IPEAK=PEAK
            IMAG(0)=IMAG(124)
800     CONTINUE
        GO TO 105

********************* END ********************
```

```
999     WRITE(*,*)'FINISHED',CHAR(7),CHAR(7),CHAR(7)
        stop
        end
**************** CUBIC SPLINE INTERPOLATION ********************

        SUBROUTINE SPLINE(MAG,MAG16)
        INTEGER*2 MAG(0:124),MAG16(1984)
        REAL MAGD2(0:124),U(124)
***** SET BOUNDARY CONDITIONS ******
        MAGD2(0)=0.
        U(1)=0.
        MAGD2(124)=0.
***** FIND SECOND DERIVATIVES ******
        DO 11 I=1,123
                P=MAGD2(I-1)/2.+2.
                MAGD2(I)=(-.5)/P
                U(I)=(3.*(MAG(I+1)-2*MAG(I)+MAG(I-1))-.5*U(I-1))/P
11      CONTINUE
        DO 12 K=123,0,-1
                MAGD2(K)=MAGD2(K)*MAGD2(K+1)+U(K)
12      CONTINUE
***** SPLINE TO INTERPOLATE ********
        DO 13 I=1,1984
                J=(I-1)/16
                IF(REAL(I-1)/16.0.EQ.J)THEN
                        MAG16(I)=MAG(J)
                ELSE
                        A=REAL((J+1)*16-I)/16.0
                        B=REAL(I-J*16)/16.0
                        MAG16(I)=A*MAG(J)+B*MAG(J+1)+(A**3-A)*MAGD2(J)+
        Z                               (B**3-B)*MAGD2(J+1)
                        ENDIF
13      CONTINUE
        RETURN
        END
************ DRAWING SUBROUTINE ***************************
        SUBROUTINE DRAW(MAG,YDIV)

        CHARACTER*1 DOT(124,10),ANS$
        CHARACTER*62 SC(4)
        INTEGER DX,DY,YDIV
        INTEGER*2 MAG(124)

40      FORMAT (A1)
50      FORMAT (2A32)

        SC(1)='0       10      20      30  '
        SC(2)='        40      50      60  < '
        SC(3)='        70      80      90      '
        SC(4)='   100 110     120     < '
```

```
************* ERASE THE SCREEN *****************
90      DO 101 DX=1,124
                DO 100 DY=1,10
                        DOT(DX,DY)=' '
100             CONTINUE
101     CONTINUE
************* WRITE POINTS *********************
        DO 160 DX=1,124
                DY=MAG(DX)/YDIV
                IF(DY.GT.10)DY=10
                IF(DY.LT.1)DY=1
                DOT(DX,DY)='*'
160     CONTINUE
        DO 200 DY=10,1,-1
                WRITE(*,*)(DOT(DX,DY),DX=1,62)
200     CONTINUE
        WRITE(*,50)SC(1),SC(2)
        DO 210 DY=10,1,-1
                WRITE(*,*)(DOT(DX,DY),DX=63,124)
210     CONTINUE
        WRITE(*,50)SC(3),SC(4)
************* CHECK GRAPH ***********************
        WRITE(*,*)'IS THE Y MAGNITUDE OKAY?(Y/N)'
        READ(*,40)ANS$
        IF(ANS$.EQ.'N'.OR.ANS$.EQ.'n')THEN
                WRITE(*,*)'INPUT A NEW DIVISOR(NOW ITS',YDIV,' ):'
                WRITE(*,*)'BIGGER DIVISOR => SMALLER AMPLITUDE'
                READ(*,*)YDIV
                GO TO 90
        ENDIF
        RETURN
        END
```

F. PROGRAM AGRAF4

This program takes the magnitude and phase from programs AMORE, AHAD, ACRID, and AGONY and breaks it into data files which can be used to make waterfall plots using MATLAB. The output file consists of 124 points (one code period) followed by a carriage return. Up to about 8200 points (65 lines) can be plotted using the command MESH.

```
*THIS PROGRAM TAKES THE OUTPUT OF THE VARIOUS DATA PROGRAMS
*(MAGNITUDE AND PHASE) AND CONVERTS IT TO A FORM USEABLE
*BY MATLAB FOR GRAPHING. A 124x65 MATRIX IS THE MAXIMUM
*OUTPUT SIZE MATLAB CAN HANDLE.

      CHARACTER*60 WORD$
      CHARACTER*20 IFILE$,MFILE$,PFILE$
      CHARACTER*6 MAT$(18)
      CHARACTER*4 MBAS$,PBAS$
      CHARACTER*1 ANS$,ANS2$
      INTEGER*4 MAG(124),PHASE(124)
      INTEGER PNUM
      DATA  MAT$/'01.MAT','02.MAT','03.MAT','04.MAT','05.MAT','06.MAT',
     Z   '07.MAT','08.MAT','09.MAT','10.MAT','11.MAT','12.MAT',
     Z   '13.MAT','14.MAT','15.MAT','16.MAT','17.MAT','18.MAT'/
      PNUM=1
35    FORMAT (124I10,1A1)
40    FORMAT (2I5)
50    FORMAT (A60)
60    FORMAT (A20)
70    FORMAT (A1)
80    FORMAT (A4)
      WRITE(*,*)'THIS PROGRAM TAKES OUTPUT MAGNITUDE AND PHASE '
      WRITE(*,*)'AND BREAKS IT INTO DATA FILES FOR PLOTTING WITH '
      WRITE(*,*)'MATLAB'
      WRITE(*,*)'INPUT FILENAME?EXAMPLE: E:B1218.MAG'
      READ(*,60)IFILE$
      WRITE(*,*)'OUTPUT FILENAME FOR MAGNITUDE?**FOUR LETTERS**'
      WRITE(*,*)' USE A BASE LIKE "B12X", THE PROGRAM WILL MAKE'
      WRITE(*,*)'IT "C:\MATLAB\B12X01.MAT", THEN "02.MAT",ETC.'
      READ(*,80)MBAS$
      WRITE(*,*)'DO YOU WANT TO OUTPUT PHASE(Y/N)?'
      READ(*,70)ANS$
      IF(ANS$.EQ.'Y'.OR.ANS$.EQ.'y')THEN
            WRITE(*,*)'OUTPUT FILENAME FOR PHASE?EXAMPLE:"PB12"'
            READ(*,80)PBAS$
            PFILE$='C:\MATLAB\'//PBAS$//MAT$(PNUM)
```

```
            OPEN(32,FILE=PFILE$)
            REWIND 32
      ENDIF
      WRITE(*,*)'WAS THIS DATA POST-PROCESSED?'
      READ(*,70)ANS2$
      OPEN(33,FILE=IFILE$,STATUS='OLD')
      MFILE$='C:\MATLAB\//MBAS$//MAT$(PNUM)
      WRITE(*,*)MFILE$
      OPEN(34,FILE=MFILE$,STATUS='UNKNOWN')
      REWIND 34
      read(33,50)WORD$
      WRITE(*,*)WORD$
      read(33,50)WORD$
      WRITE(*,*)WORD$
      IF(ANS2$.EQ.'Y'.OR.ANS2$.EQ.'y')THEN
            READ(33,50)WORD$
            WRITE(*,*)WORD$
      ENDIF
      WRITE(*,*)'HOW MANY 1.9375 SEC GROUPS?(MAX=65)'
      READ(*,*)K
      WRITE(*,*)'SKIP?(1 FOR ALL, 3 FOR EVERY THIRD, ETC)'
      READ(*,*)KK
      write(*,*)'skip a few at the beginning?'
      read(*,*)ISKIP
      IF(ISKIP.EQ.0)GOTO 410
      DO 400 J=1,ISKIP
            READ(33,50)
400   CONTINUE
410   CONTINUE
      DO 100 J=1,K*KK
            DO 10 N=1,124
                  READ(33,40,end=999)MAG(N),PHASE(N)
                  MAG(N)=MAG(N)**2
10          continue
            IF((J/KK).NE.(REAL(J)/REAL(KK)))GOTO 100
            WRITE(34,35)(MAG(N),N=1,124),CHAR(13)
            IF(ANS$.EQ.'Y'.OR.ANS$.EQ.'y')THEN
                  WRITE(32,35)(PHASE(N),N=1,124),CHAR(13)
            ENDIF
100   continue
      CLOSE (32)
      CLOSE (34)
      PNUM=PNUM+1
      MFILE$='C:\MATLAB\//MBAS$//MAT$(PNUM)
      WRITE(*,*)MFILE$
      OPEN(34,FILE=MFILE$)
      REWIND 34
      IF(ANS$.EQ.'Y'.OR.ANS$.EQ.'y')THEN
            PFILE$='C:\MATLAB\//PBAS$//MAT$(PNUM)
            OPEN(32,FILE=PFILE$)
            REWIND 32
```

```
      ENDIF
      IF(PNUM.LT.18)GO TO 410
999         continue
            WRITE(*,*)CHAR(7),CHAR(7)
      stop
      end
```

G. PROGRAM AGRAF5

This program takes the magnitude and phase from programs AMORE, AHAD, ACRID, and AGONY and breaks it into data files which can be used to make waterfall plots using SURFER. This program makes files of type *.GRD, as would be put together by the routine GRID. The output file consists of all the data points strung together with a header telling the program how the rows and columns are separated. Up to about 10000 points (80 lines) can be plotted using the routine SURF.

```
*THIS PROGRAM TAKES THE OUTPUT OF THE VARIOUS DATA PROGRAMS
*(MAGNITUDE AND PHASE) AND CONVERTS IT TO A FORM USEABLE
*BY SURFER FOR GRAPHING. A 124x80 MATRIX IS THE MAXIMUM
*OUTPUT SIZE SURFER CAN HANDLE.

      CHARACTER*60 WORD$
      CHARACTER*20 IFILE$,MFILE$,PFILE$
      CHARACTER*6 MAT$(18)
      CHARACTER*4 MBAS$,PBAS$
      CHARACTER*1 ANS$,ANS2$
      INTEGER*4 MAG(124),PHASE(124)
      INTEGER PNUM
      DATA  MAT$/'01.DAT','02.DAT','03.DAT','04.DAT','05.DAT','06.DAT',
     Z   '07.DAT','08.DAT','09.DAT','10.DAT','11.DAT','12.DAT',
     Z   '13.DAT','14.DAT','15.DAT','16.DAT','17.DAT','18.DAT'/
      PNUM=1
30    FORMAT (I3,2X,I3,2X,I10)
35    FORMAT (124I10,1A1)
40    FORMAT (2I5)
50    FORMAT (A60)
60    FORMAT (A20)
70    FORMAT (A1)
80    FORMAT (A4)
      WRITE(*,*)'THIS PROGRAM TAKES OUTPUT MAGNITUDE AND PHASE '
      WRITE(*,*)'AND BREAKS IT INTO DATA FILES FOR PLOTTING WITH '
      WRITE(*,*)'SURFER'
      WRITE(*,*)'INPUT FILENAME?EXAMPLE: D:B1218.MAG'
      READ(*,60)IFILE
      WRITE(*,*)'OUTPUT FILENAME FOR MAGNITUDE?**FOUR LETTERS**'
      WRITE(*,*)'JUST USE A BASE LIKE "B12X", THE PROGRAM WILL '
      WRITE(*,*)'MAKE IT "E:\SURFER\B12X01.DAT", THEN "02.DAT",ETC.'
      READ(*,80)MBAS$
      WRITE(*,*)CHAR(7)
      WRITE(*,*)'CERIFY THAT THE DISK WITH SURFER IS IN DRIVE E!'
      WRITE(*,*)' '
```

```
      WRITE(*,*)'DO YOU WANT TO OUTPUT PHASE(Y/N)?'
      READ(*,70)ANS$
      IF(ANS$.EQ.'Y'.OR.ANS$.EQ.'y')THEN
            WRITE(*,*)'OUTPUT FILENAME FOR PHASE?EXAMPLE:"PB12"'
            READ(*,80)PBAS$
            PFILE$='E:\SURFER\//PBAS$//MAT$(PNUM)
            OPEN(32,FILE=PFILE$)
            REWIND 32
      ENDIF
      WRITE(*,*)'WAS THIS DATA POST-PROCESSED?'
      READ(*,70)ANS2$
      OPEN(33,FILE=IFILE$,STATUS='OLD')
      MFILE$='E:\SURFER\//MBAS$//MAT$(PNUM)
      WRITE(*,*)MFILE$
      OPEN(34,FILE=MFILE$,STATUS='UNKNOWN')
      REWIND 34
      read(33,50)WORD$
      WRITE(*,*)WORD$
      read(33,50)WORD$
      WRITE(*,*)WORD$
      IF(ANS2$.EQ.'Y'.OR.ANS2$.EQ.'y')THEN
            READ(33,50)WORD$
            WRITE(*,*)WORD$
      ENDIF
      WRITE(*,*)'HOW MANY 1.9375 SEC GROUPS?(MAX=65)'
      READ(*,*)K
      WRITE(*,*)'SKIP?(1 FOR ALL, 3 FOR EVERY THIRD, ETC)'
      READ(*,*)KK
      write(*,*)'skip a few at the beginning?'
      read(*,*)ISKIP
      IF(ISKIP.EQ.0)GOTO 410
      DO 400 J=1,ISKIP
            READ(33,50)
400   CONTINUE
410   CONTINUE
      DO 100 J=1,K*KK
            DO 10 N=1,124
                  READ(33,40,end=999)MAG(N),PHASE(N)
                  MAG(N)=MAG(N)**2
10          continue
            IF((J/KK).NE.(REAL(J)/REAL(KK)))GOTO 100
            DO 99 N=1,124
                  WRITE(34,30)(J/KK),N,MAG(N)
                  IF(ANS$.EQ.'Y'.OR.ANS$.EQ.'y')THEN
                        WRITE(32,35)(J/KK),N,PHASE(N)
                  ENDIF
99          CONTINUE
100   continue
      CLOSE (32)
      CLOSE (34)
      PNUM=PNUM+1
```

```
      MFILE$='E:\SURFER\//MBAS$//MAT$(PNUM)
      WRITE(*,*)MFILE$
      OPEN(34,FILE=MFILE$)
      REWIND 34
      IF(ANS$.EQ.'Y'.OR.ANS$.EQ.'y')THEN
            PFILE$='E:\SURFER\//PBAS$//MAT$(PNUM)
            OPEN(32,FILE=PFILE$)
            REWIND 32
      ENDIF
      IF(PNUM.LT.18)GO TO 410
999         continue
            WRITE(*,*)CHAR(7),CHAR(7)
      stop
      end
```

APPENDIX D

Additional Data for Station J

A. MATCHED-FILTERED ACOUSTIC SIGNAL

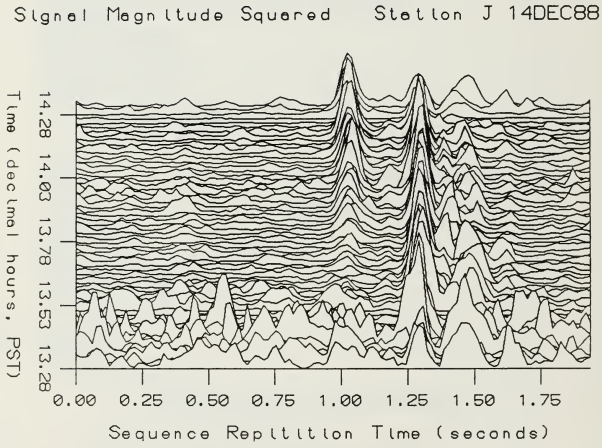


Figure 26: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1317 to 1419 14DEC88. High ambient noise at the start is from the R/V Point Sur after deploying buoy.

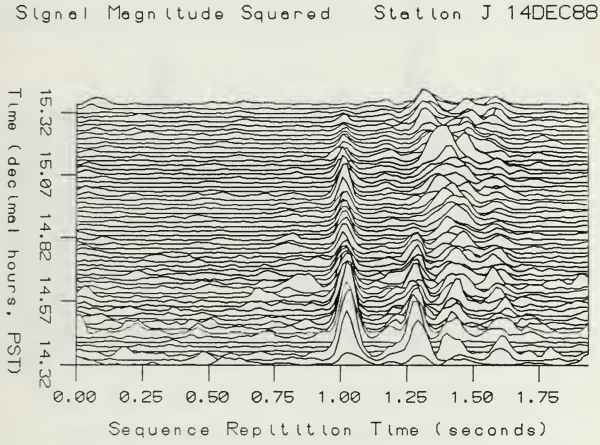


Figure 27: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1419to 1521 14DEC88.

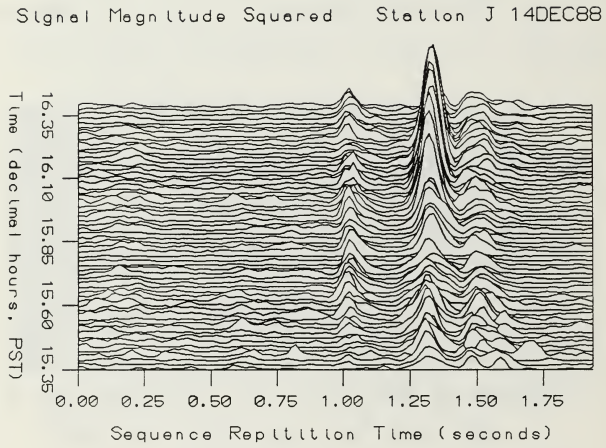


Figure 28: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1521 to 1623 14DEC88.

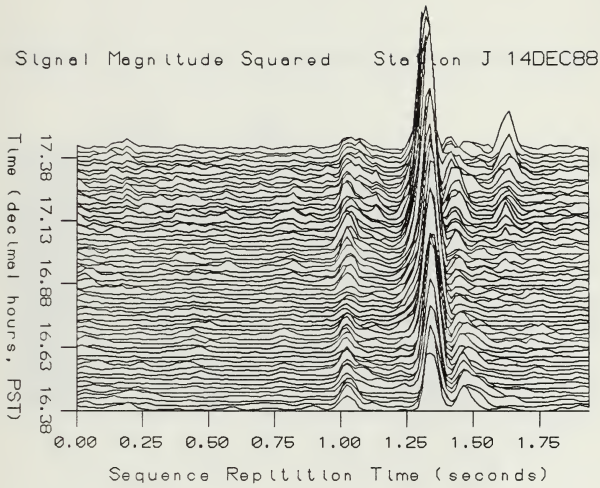


Figure 29: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1623 to 1725 14DEC88.

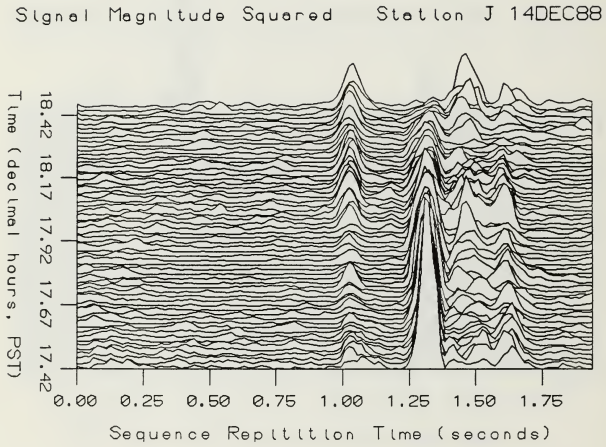


Figure 30: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1725 to 1827 14DEC88.

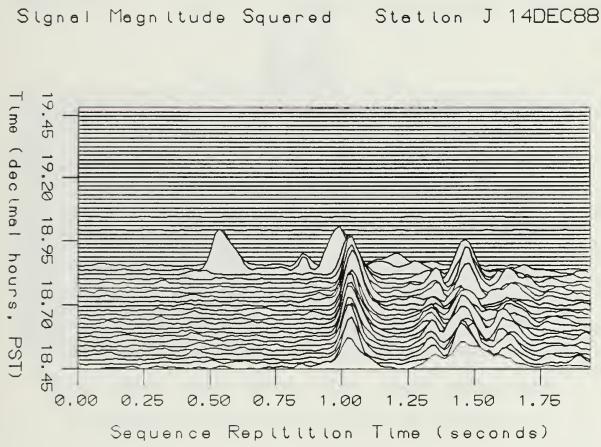


Figure 31: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1827 to 1929 14DEC88. Signal cutoff is due to tape change.

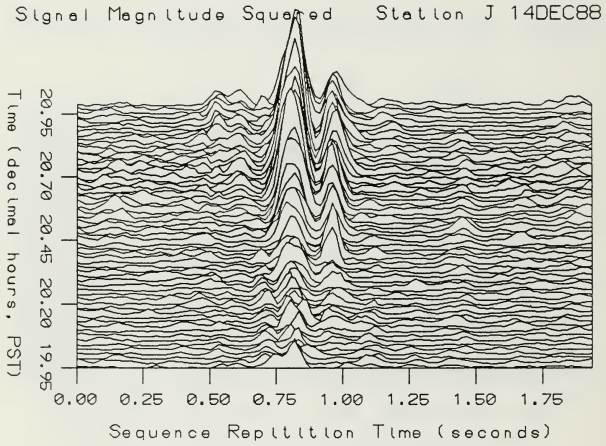


Figure 32: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1957 to 2059 14DEC88. The previous hour is included as Figure 12 on page 58. Note that the arrival structure is shifted for data from a new tape.

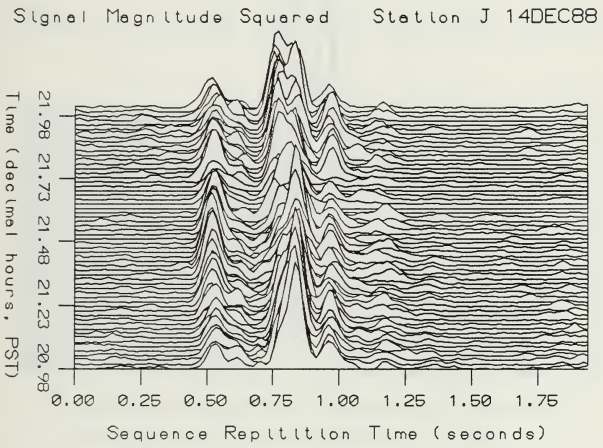


Figure 33: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 2059 to 2201 14DEC88.

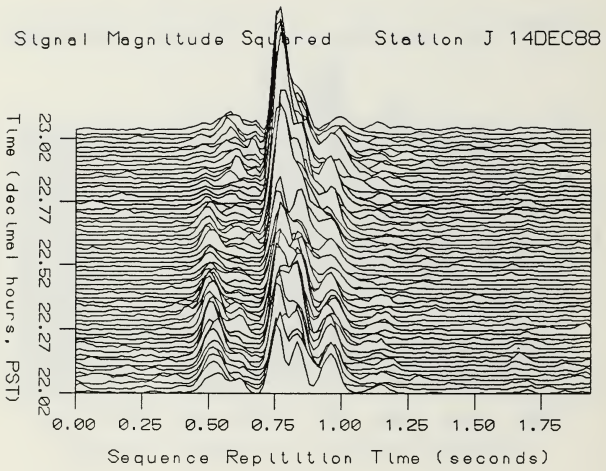


Figure 34: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 2201 to 2303 14DEC88.

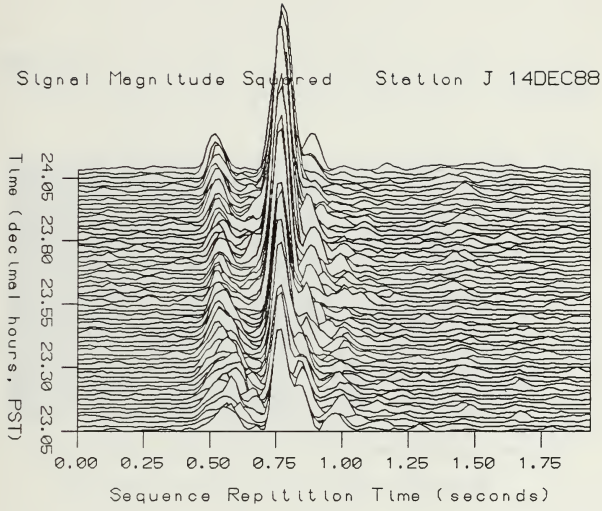


Figure 35: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 2303 14DEC88 to 0005 15DEC88.

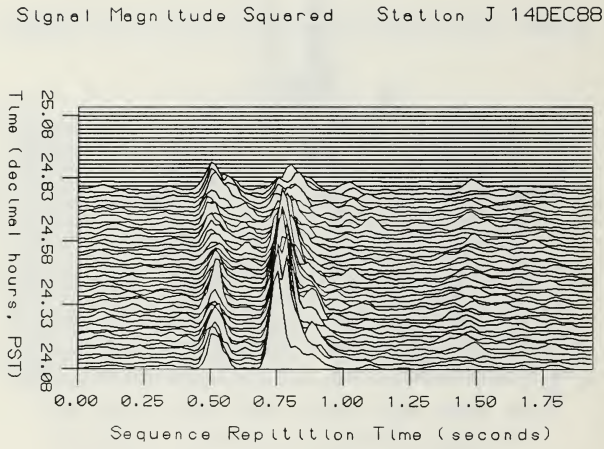


Figure 36: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0005 to 0107 15DEC88. Note that computer generated time scale is extended past 0000 for convenience in processing. The reason for signal cutoff is that the end of the tape was reached.

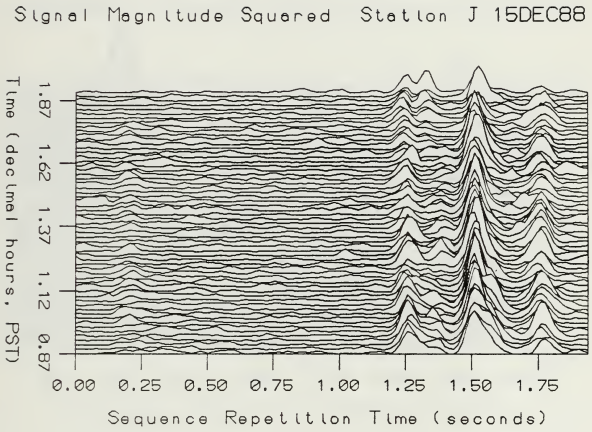


Figure 37: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0052 to 0154 15DEC88. Note that the arrival structure is shifted because of the start of a new tape.

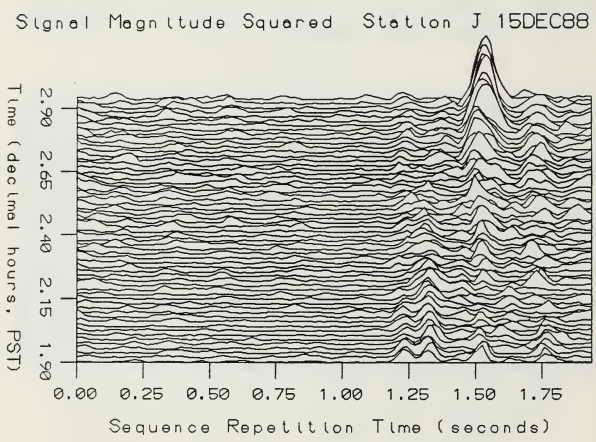


Figure 38: Tomographic signal, coherently averaged 16 timesthen magnitude squared. Station J, 0154 to 0256 15DEC88.

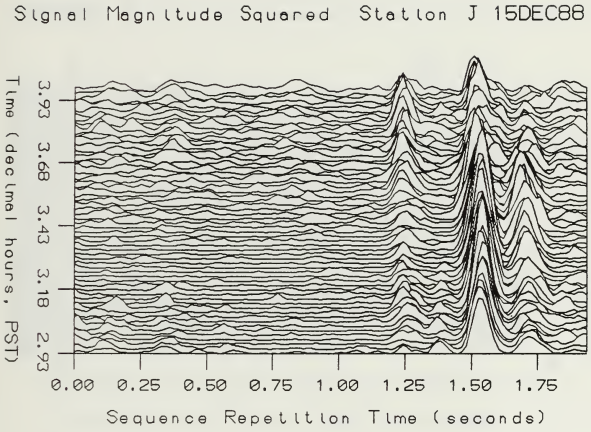


Figure 39: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0256 to 0358 15DEC88.

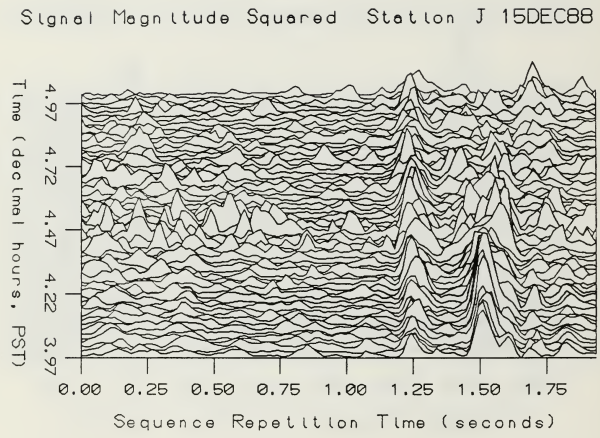


Figure 40: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0358 to 0500 15DEC88. High scattering and ambient noise were present at this time because of high winds (the worst windstorm of the year to hit the central California coast).

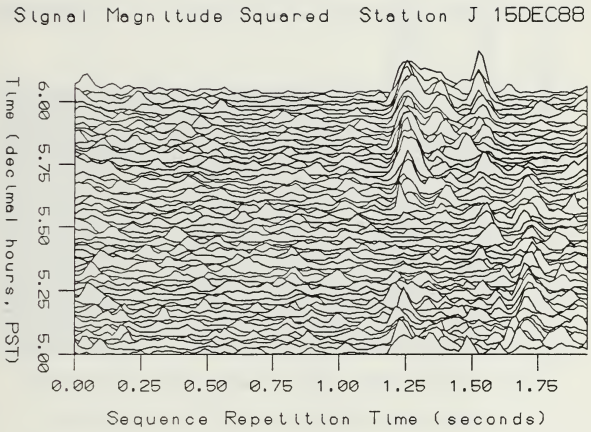


Figure 41: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0500 to 0602 15DEC88. High ambient noise and high scattering continue from windstorm.

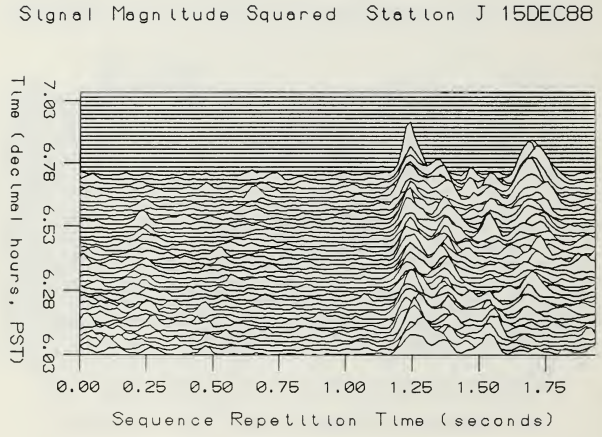


Figure 42: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0602 to 0704 15DEC88. The reason for signal cutoff is that the end of the tape was reached.

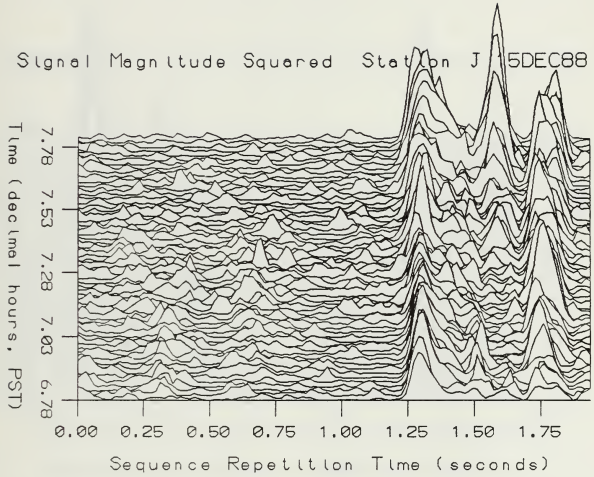


Figure 43: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0647 to 0749 15DEC88. The reason for the increased amplitude is unknown. Note that the arrival structure is shifted at the start of the new tape.

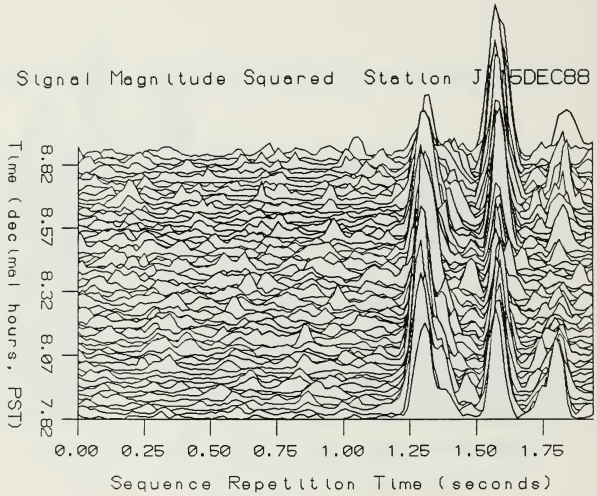


Figure 44: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0749 to 0851 15DEC88.

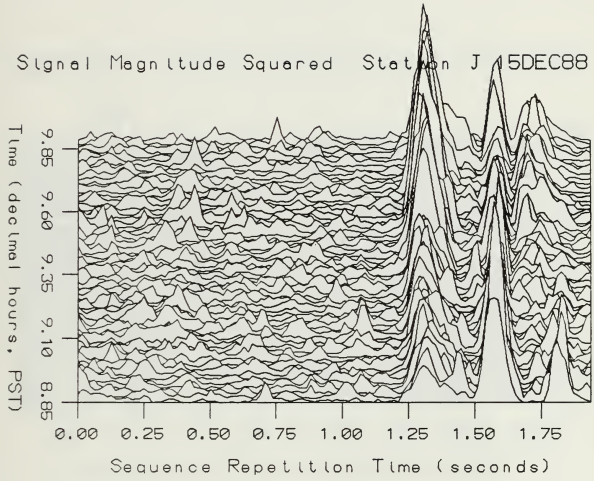


Figure 45: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0851 to 0953 15DEC88.

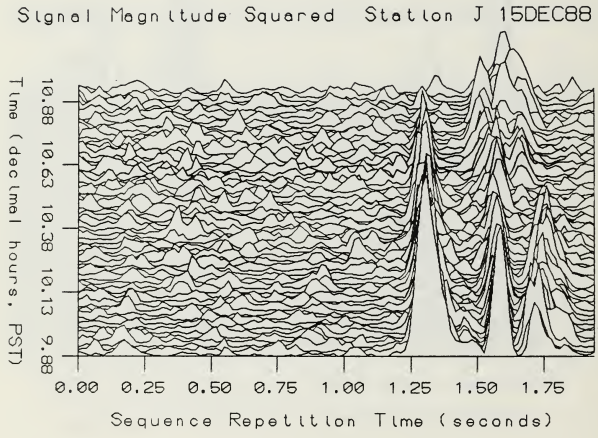


Figure 46: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 0953 to 1055 15DEC88.

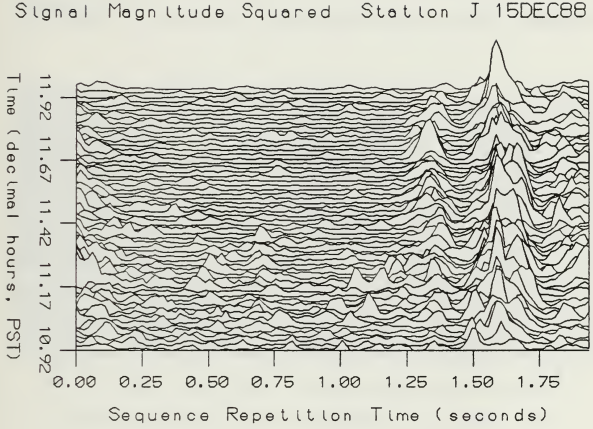


Figure 47: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1055 to 1157 15DEC88.

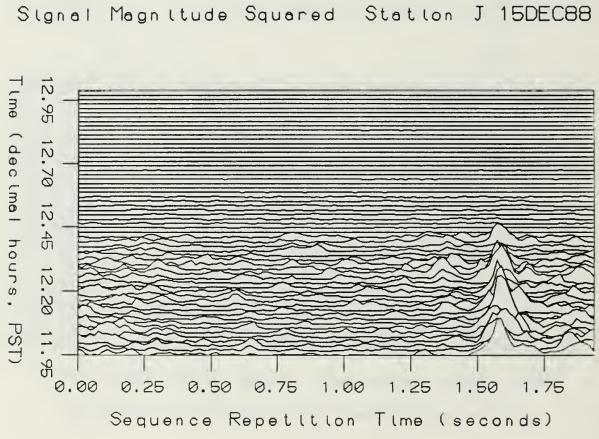


Figure 48: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1157 to 1259 15DEC88. The reason for the signal cutoff is that the end of the tape was reached.

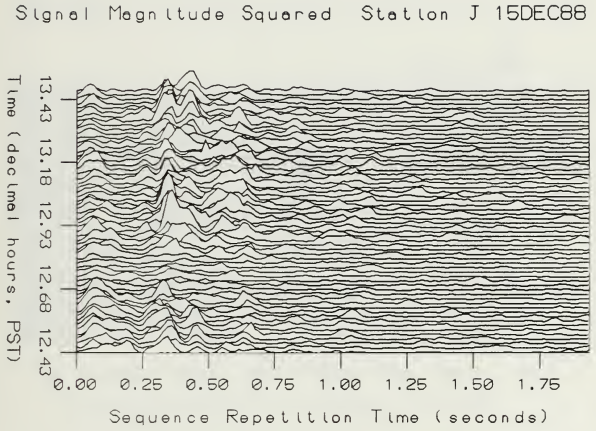


Figure 49: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1226 to 1328 15DEC88. Note that the arrival structure is shifted at the start of the new tape.

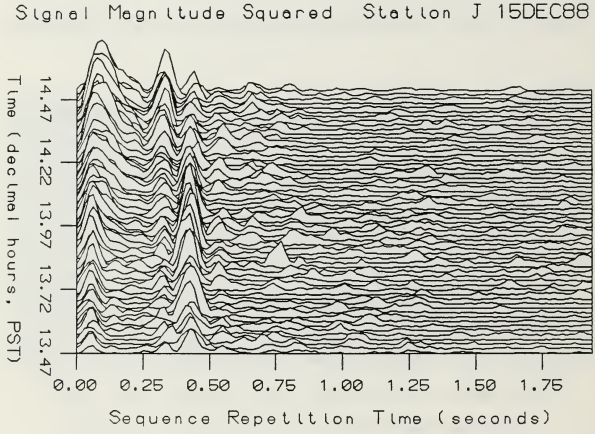


Figure 50: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1328 to 1430 15DEC88.

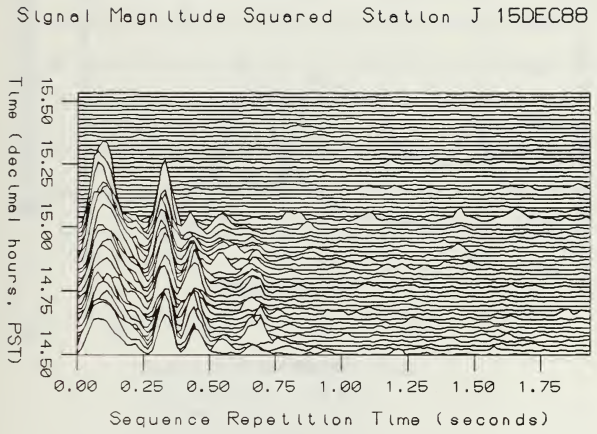


Figure 51: Tomographic signal, coherently averaged 16 times then magnitude squared. Station J, 1430 to 1532 15DEC88. Signal cutoff is due to buoy failure.

B. ARRIVAL TIME AND SURFACE WAVE SPECTRA

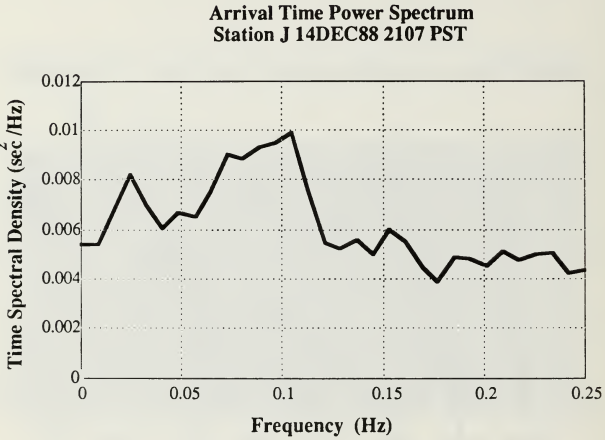
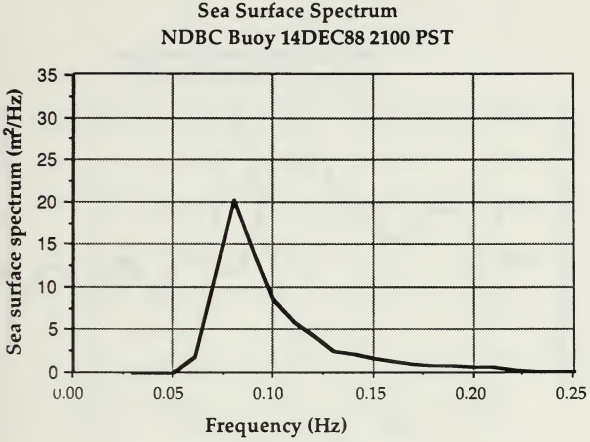


Figure 52: Arrival time power spectrum for Station J. Spectrum from 2.2 hours of Arrival Time Series, 2001 to 2213 14DEC88 PST.



Significant Wave Height 3.54 m
Average Period 9.11 sec
Dominant Period 12.50 sec
Dominant Direction 314 N

Figure 53: Surface wave power spectrum in Monterey Bay. Data is from the NDBC buoy southwest of Santa Cruz, 2100 14DEC88 PST.

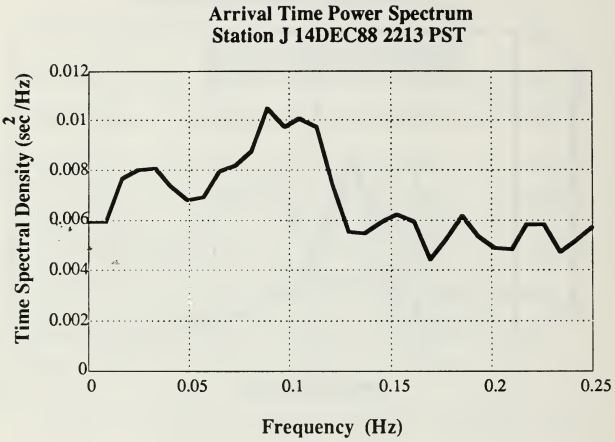
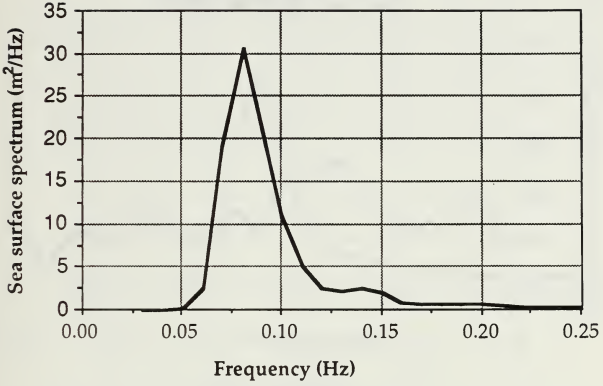


Figure 54: Arrival time power spectrum for Station J. Spectrum from 2.2 hours of Arrival Time Series,2107 to 2319 14DEC88 PST.



Significant Wave Height 4.10 m
Average Period 9.67 sec
Dominant Period 12.50 sec
Dominant Direction 308 N

Figure 55: Surface wave power spectrum in Monterey Bay. Data is from the NDBC buoy southwest of Santa Cruz, 2200 14DEC88 PST.

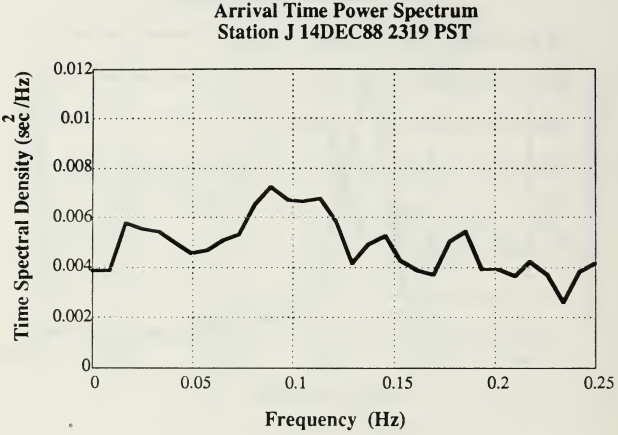
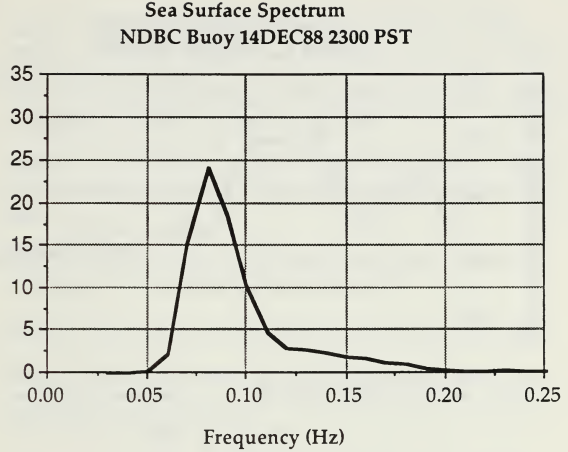


Figure 56: Arrival time power spectrum for Station J. Spectrum from 1.9 hours of Arrival Time Series,2213 14DEC88 to 0005 15DEC88 PST.



Significant Wave Height 3.85 m
Average Period 9.36 sec
Dominant Period 12.50 sec
Dominant Direction 321 N

Figure 57: Surface wave power spectrum in Monterey Bay. Data is from the NDBC buoy southwest of Santa Cruz, 2300 14DEC88 PST.

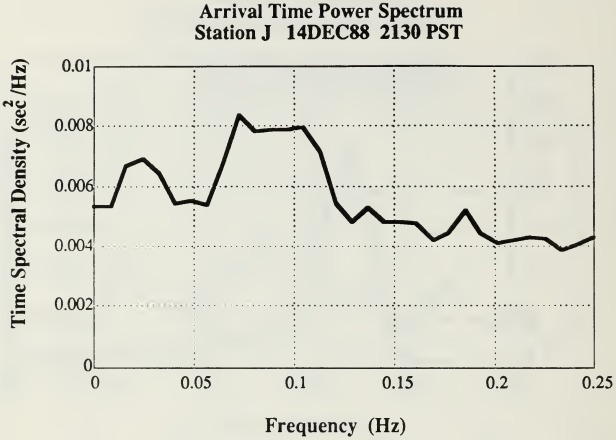


Figure 58: Arrival time power spectrum for Station J. This spectrum was generated using the segmented Fast Fourier Transform Method on the data from an entire tape (the maximum length data series without sychronization between tapes).

References

1. Spindel, R.C., "Ocean Acoustic Tomography: A Review," *Current Practices and New Technology in Ocean Engineering*, v. 11, pp. 7 - 13, 1986.

2. The Ocean Tomography Group, "A Demonstrationof Ocean Acoustic Tomography," *Nature*, v. 299, pp. 121 - 125, 1982.

3. Munk, W. and Wunsch, C., "Ocean Acoustic Tomography: A Scheme for Large Scale Monitoring," *Deep-Sea Research*, v. 26A, pp. 123 - 161, 1979.

4. Flatté, Stanley, and others, *Sound Transmission through a Fluctuating Ocean*, pp. 3 - 61, Cambridge University Press, 1979.

5. Bascom, William, *Waves and Beaches*, pp. 5 - 73, Anchor Books, 1980.

6. Rowan, Theresa M., *Monterey Bay Acoustic Tomography: Ray Tracing and Environmental Assessment*, MS Thesis, Naval Postgraduate School, Monterey, CA, September 1988.

7. Conversation and data exchange concerning Monterey Bay tides between W. W. Broenkow, oceanographer, Moss Landing Marine Laboratories, Moss Landing, CA, and author on 26 April 1989.

8. Clay, C.S. and Medwin, H., *Acoustical Oceanography: Principles and Applications*, pp. 11 - 16, John Wiley & Sons, 1977.

9. Miller, J.H., Lynch, J.F., and Chiu, C.S., "Estimation of Sea Surface Spectra using Acoustic Tomography", *J. Acoust. Soc. Am.*, 86(1) pp. 326-345, July 1989.

10. Shepard, F.P., "Progress of Internal Waves Along Submarine Canyons," *Marine Geology*, v. 19, pp. 131 - 138, 1975.

11. Shea, R.E. and Broenkow, W.W., "The Role of Internal Tides in the Nutrient Enrichment of Monterey Bay, California," *Estuarine, Coastal, and Shelf Science*, v. 15, pp. 56 - 66, 1982.

12. Sparton Defense Electronics ASW Technical Center, *Operation Manual for Mobile Inshore Underwater Warfare Moored Sensor Buoy AN/SSQ-58 (7938-OPMAN-1)*, Sparton Corporation, Jackson, MI.

13. Naval Air Systems Command, *Sonobuoy Instructional Manual (NAVAIR 28-SSQ-500-1)*, Naval Weapons Support Center, Crane, IN.

14. Steele, Kenneth E., Lau, J.C., and Hsu, Y.L., "Theory and Application of Calibration Techniques for an NDBC Directional Wave Measurements Buoy," *IEEE Journal of Oceanic Engineering*, v. OE-10, no. 4, October 1985.

15. Collecte Localisation Satellites, *ARGOS Location and Data Collection System*, C.L.S., Toulouse, France, July 1987.

16. Kinsler, L.E., and others, *Fundamentals of Acoustics*, 3rd ed., pp. 117 - 120, John Wiley & Sons, 1982.

17. Menke, William, *Geophysical Data Analysis: Discrete Inverse Theory*, pp. 35 -160, Academic Press, 1984.

18. Spindel, Robert C., "Signal Processing in Ocean Tomography," *Adaptive Methods in Underwater Acoustics*, ed. H.G. Urban, pp. 687-710, D. Reidel Publishing Company, 1985.

19. Van Trees, Harry L., *Detection, Estimation and Modulation Theory, Part I*, pp. 273-287, John Wiley & Sons, 1968.

20. Press, William H. and others, *Numerical Recipes, The Art of Scientific Computing*, pp. 86-89, Cambridge University Press, 1986.

21. Bendat, Julius S. and Piersol, Allan G., *Random Data: Analysis and Measurement Procedures*, pp. 314 - 365, John Wiley & Sons, 1971.

22. Deck Log of the Research Vessel Point Sur, 12 to 16 December 1988.

23. Lynch, Peter J., *Computer Simulation of Gold Code Phase Modulation in Ocean Acoustic Tomography*, MS Thesis, Naval Postgraduate School, Monterey, CA, June 1989.

24. Ziemer, Rodger E. and Peterson, Rodger l., *Digital Communications and Spread Spectrum Systems*, pp. 385-404, Macmillan Publishing Company, 1985.

25. Birdsall, Ted, letter dated 27 May 1987, " My Introduction to Hadamard Processes for M-Sequences."

26. Cohn, Martin and Lempel, Abraham, "On Fast M-Sequence Transforms," *IEEE Transactions on Information Theory*, pp. 135 - 137, January 1976.

27. Borish, Jeffrey and Angell, James, "An Efficient Algorithm for Measuring the Impulse Response Using Pseudorandom Noise,"*J. Audio Eng. Soc.*, Vol. 31, No. 7, pp. 478 - 488, July/August 1983.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Prof. James H. Miller, Code 62Mr Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943	12
4.	Prof. Ching-Sang Chiu, Code 68Ci Department of Oceanography Naval Postgraduate School Monterey, CA 93943	1
5.	Dr. James F. Lynch Department of Applied Ocean Physics and Engineering Woods Hole Oceanographic Institution Woods Hole, MA 02543	1
6.	LT Robert C. Dees, USN 207 Lauber Lane Derby, KS 67037	2

7. Dr. R. K. R. Cook 1
Strong Hall, Room 108
University of Kansas
Lawrence, KS 66045

8. Dr. Kurt Metzger, Jr. 1
Communications and Signal Processing Laboratory
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48109

9. Dr. Marshall Orr 1
Office of Naval Research, Code 11250A
800 North Quincy Street
Arlington, VA 22217-5000

10. Dr. Robert C. Spindel 1
Director, Applied Physics Laboratory
University of Washington
1013 Northeast 40th Street
Seattle, WA 98105

11. Floyd L. York, Code 68Yk 1
Department of Oceanography
Naval Postgraduate School
Monterey, CA 93943

12. Sönke Paulsen, Code 62Pa 1
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, CA 93943

13. Mrs.Theresa Rowan,Code 742 — 1
Naval Training Systems Center
12350 Research Parkway
Orlando, FL 32826

14. Dr. Warren Denner — 1
SAIC
205 Montecito Avenue
Monterey, CA 93940

15. LT(N) Edwin K. Chaulk, CDN Navy — 1
Weapons Curricular Office, WT91
Naval Postgraduate School
Monterey, CA 93943

16. Dr. John Stockhausen — 1
Defense Research Establishment Atlantic
Grove Street, P.O. Box 1012
Dartmouth, Nova Scotia, Canada B2Y3Z7

Thesis
D234493 Dees
c.1 Signal processing and preliminary results in the 1988 Monterey Bay Tomography Experiment.